A Parallel High-Order Accurate Finite Element Nonlinear Stokes Ice Sheet Model and Benchmark Experiments

Wei Leng,[1] Lili Ju,[2] Max Gunzburger,[3] Stephen Price,[4] and Todd Ringler[4]

Abstract. The numerical modeling of glacier and ice sheet evolution is a subject of growing interest, in part because of the potential for models to inform estimates of global sea-level change. This paper focuses on the development of a numerical model that determines the velocity and pressure fields within an ice sheet. Our numerical model features a high-fidelity mathematical model involving the nonlinear Stokes system and combinations of no-sliding and sliding basal boundary conditions, high-order accurate finite element discretizations based on variable resolution grids, and highly scalable parallel solution strategies, all of which contribute to a numerical model that can achieve accurate velocity and pressure approximations in a highly efficient manner. We demonstrate the accuracy and efficiency of our model by analytical solution tests, established ice sheet benchmark experiments, and comparisons with other well-established ice sheet models.

1. Introduction

Over the few past decades, numerical simulations of glacier and ice sheet evolution have been a subject of growing interest, in part due to their potential for informing estimates of global sea-level change [*Alley et al.*, 2005; *Gregory and Huybrechts*, 2006; *Bamber et al.*, 2007; *Shepherd and Wingham*, 2007]. Among the different approaches available, the three-dimensional nonlinear Stokes equations (hereafter simply referred to as the Stokes equations) provide the most accurate and complete description of momentum balance for modeling the flow of land ice [*Blatter*, 1995; *Hindmarsh*, 2004; *Le Meur et al.*, 2004; *Pattyn*, 2003; *Pattyn et al.*, 2008; *Gagliardini et al.*, 2008]. Stokes ice sheet modeling is required to take full advantage of recent advances in both the coverage and spatial resolution of ice sheet geometry data. It is also required to provide accurate results in regions where steep topography and low aspect ratios are inherent in the problem, where the aspect ratio is defined to be the ratio of the horizontal to the vertical extents of the ice sheet. For example, for the Jakobshavn outlet glacier trough in Greenland, a recent high-resolution data set suggests basal topography with slopes as large as 45 degrees [*Plummer and van der Veen*, 2011]. For slopes of this magnitude, the sliding speeds predicted by the Stokes-flow model and a first-order model differ by a factor of 4, whereas those for the Stokes-flow model and a zeroth-order ("shallow ice") model differ by a factor of 8 [*Dukowicz et al.*, 2011]. Stokes ice sheet modeling is also needed to take full advantage of more complex basal boundary conditions, for example any boundary condition that does not assume a hydrostatic pressure distribution at the ice-bed interface.

A number of recent studies demonstrate the limitations and errors inherent when lower-order approximations, i.e., simplifications, of the Stokes equations are applied to realistic problems of land ice flow. These include, for example, applications in areas where the bedrock topography is steep, the ice thickness changes considerably over short length scales, or the aspect ratio of the domain under consideration or of important local features is of order one [*Hindmarsh*, 2004; *Le Meur et al.*, 2004; *Zwinger et al.*, 2007; *Pattyn et al.*, 2008]; in studies of ice sheet grounding line behavior, where lower-order approximations may be inaccurate due to complicated geometries and changes in the amount of basal slip [*Durand et al.*, 2009; *Morlighem et al.*, 2010]; and at ice sheet flow divides, where flow modeling is crucial for interpreting ice-core derived paleoclimate records [*Price et al.*, 2007; *Neumann et al.*, 2008] and where lower-order approximations are not sufficient to accurately describe the flow field [*Martin et al.*, 2009]. Further, models based on reductions of the Stokes equations often have complex boundary conditions that are simplifications (based on integral averages and scaling analyses) of the boundary conditions one is able to impose for the Stokes formulation. The boundary conditions for the Stokes formulation are directly tied to full, three-dimensional stress and velocity fields and are therefore more accurate than those applied under simplified formulations.

In addition to the importance of solving the Stokes equations, studies show that high-quality, adaptive, variable resolution meshes are necessary to manage computational costs whereas accurately capturing essential dynamic behaviors, such as the spatial variability in flow velocities, sensitivities to meltwater and subglacial sediments, and behavior at ice sheet grounding lines [*Schoof*, 2007; *Nick et al.*, 2009; *Goldberg et al.*, 2009; *Durand et al.*, 2009; *Morlighem et al.*, 2010]. Mesh adaptivity is also useful for incorporation of better geometric resolution of ice sheet boundaries, which calls for fine grids near the boundary but not necessarily in the interior. Moreover, as the complexity of physical processes included in large-scale models is expected to increase in the future, for example through the inclusion of evolutionary basal hydrology and basal process models, the capability to focus resolution locally and adaptively will become increasingly important for both obtaining accurate solutions and for reducing the computational costs associated with a three-dimensional Stokes solution (as compared to the use of a quasi-uniform mesh).

Whereas a number of numerical methods, such as finite differences, finite volumes, and finite elements, have

[1] State Key Laboratory of Scientific and Engineering Computing, Chinese Academy of Sciences, Beijing 100190, China. Email: wleng@lsec.cc.ac.cn.

[2] Department of Mathematics, University of South Carolina, Columbia, SC 29208, USA. Email: ju@math.sc.edu.

[3] Department of Scientific Computing, Florida State University, Tallahassee, FL 32306, USA. Email: gunzburg@fsu.edu.

[4] Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM 87545. USA. Email: sprice,ringler@lanl.gov.

Copyright 2011 by the American Geophysical Union.
0148-0227/11/$9.00

been used for ice sheet modeling, it is not always obvious which of these methods is best suited for the considerations discussed above, particularly when considering large-scale, high-resolution, time-evolving simulations of realistic glaciers and ice sheets. High-order accurate numerical discretizations and highly scalable parallel solvers become not only desirable, but necessary; the former can greatly reduce the size of the resulting discrete system while maintaining comparable solution accuracy whereas the latter can take full advantage of modern high-performance computing architectures. We note that "high-order accurate" refers to accuracy of the discretization as opposed to "higher-order model" [*Pattyn*, 2003] for ice sheet modeling which refers to a simplification of the Stokes model.

Finite element methods are, in general, more suitable, compared to other methods, for problems with complex geometries and complex boundary conditions. Highly variable resolution grids that conform to the boundary are needed, and finite element methods that do not suffer a loss of accuracy on such meshes are relatively easy to design. For ice sheet modeling this is especially important because the boundary geometry can be very irregular and made up of many short segments with both acute and obtuse angles. Also, not only are there Neumann-type boundary conditions at, e.g., the ice-atmosphere boundary, but the boundary condition at the ice-bedrock boundary can involve a very complex, nonlinear sliding condition. These boundary conditions can be incorporated into variational formulations of the problem (that form the basis of finite element methods) in a rather straightforward manner, as discussed further in Section 2.3. Thus, in this paper, we focus on the development of an accurate, efficient, parallel Stokes solver for large-scale ice sheet modeling using high-order-accurate finite element methods.

In a previous paper by *Zhang et al.* [2011], the authors also considered a three-dimensional, finite element discretization of the nonlinear Stokes equations for ice sheets. The current paper differs in several important ways. First, in *Zhang et al.* [2011], only second-order accurate, piecewise-linear finite element (P1-P1) discretizations were considered, whereas here we consider third-order accurate discretizations for the velocity field. In Section 4.1, we show that there are clear advantages to the latter. Second, in *Zhang et al.* [2011], a penalty method was used to stabilize the finite element method; this introduces a non-physical penalty parameter into the problem that not only affects accuracy, but also the conditioning of the linear systems to be solved. The finite element method used here does not require stabilization and so there is no need to introduce a penalty term for stabilization. Third, in *Zhang et al.* [2011], only the no-sliding condition is used at the basal boundary whereas here we consider sliding boundary conditions as well. Thus, the current paper represents a significant improvement over the model presented in *Zhang et al.* [2011]. As is demonstrated below, all of these improvements result in lower computational cost and higher accuracy in realistic glacier and ice sheet simulations.

Finite element discretizations were used for glaciology in *Truffer et al.* [2003]; an important advance presented in this paper is the use of these discretization techniques on a very large spatial scale. Specifically, we explore the use of quadratic finite element velocity approximations. We note that penalization is not the only means for stabilizing the use of linear velocity approximations. The use of finer velocity grids (relative to the pressure grid) or "bubble functions" to enhance the velocity space can also be used for stabilization. Such enhancements increase the number of degrees of freedom for the velocity (so that they are approximately equivalent to the Taylor-Hood element we use) without a commensurate increase in accuracy. A final note is that the use of high-order accurate finite element discretizations often requires a more costly matrix assembly, mainly because of high-order accurate quadrature rules. However, for the Stokes problem, and assuming that one does not want to use penalization-based stabilization (e.g., because of the difficulties and inaccuracy it can cause solvers), the other available stabilization approaches for linear velocity approximations involve velocity grid refinement or higher degree polynomial bubble functions, both of which also require more quadrature points during the assembly process.

The paper is organized as follows. In Section 2, we provide a short review of the nonlinear Stokes ice sheet model, including the governing equations and boundary conditions. We also define a variational formulation of the Stokes system which is used in Section 3.1 to define high-order accurate finite element discretizations and to define the corresponding discretized systems. During the development of the high-order accuracy finite element model in Section 3, we also discuss the process for generation of variable resolution, anisotropic, tetrahedral and layered grids (Section 3.1.1), the preconditioning techniques, and parallel solver implementation used in our solution process (Section 3.2). In Section 4, we first demonstrate the high-order numerical accuracy of our Stokes solver through an example with an analytic (manufactured) solution. We then provide a comparison between our model results and those from the ISMIP-HOM benchmark experiments [*Pattyn et al.*, 2008]. The performance and parallel scalability of our model are studied in Section 5 and some concluding remarks are given in Section 6.

2. The nonlinear Stokes ice sheet model

2.1. Governing equations

The dynamical behavior of ice sheets is modeled by the Stokes equations for an incompressible viscous fluid with a nonlinear rheology, i.e., a nonlinear constitutive law is assumed. Letting $[0, t_{max}]$ denote the time interval of interest and Ω_t the three-dimensional spatial domain occupied by the ice sheet, we then have

$$\nabla \cdot \sigma + \rho \mathbf{g} = 0 \quad \text{in } \Omega_t \times [0, t_{max}], \qquad (1)$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega_t \times [0, t_{max}], \qquad (2)$$

where $\mathbf{u} = (u_x, u_y, u_z)^T$ denotes the velocity, σ the full stress tensor, ρ the density of ice, and $\mathbf{g} = (0, 0, -\|\mathbf{g}\|)$ the gravitational acceleration. The stress tensor σ can be decomposed in terms of the deviatoric stress τ and the isotropic pressure p as

$$\sigma = \tau - p\mathbf{I} \qquad \text{or} \qquad \sigma_{ij} = \tau_{ij} - p\delta_{ij}, \qquad (3)$$

where $p = -\frac{1}{3}tr(\sigma)$, δ_{ij} denotes the Kronecker delta tensor, and $\mathbf{I}$ the unit tensor. Combining (1) and (3), we obtain the *instantaneous* momentum balance equation

$$-\nabla \cdot \tau + \nabla p = \rho \mathbf{g} \qquad \text{in } \Omega_t \times [0, t_{max}]. \qquad (4)$$

The strain-rate tensor $\dot{\varepsilon}_{\mathbf{u}}$ is defined as

$$(\dot{\varepsilon}_{\mathbf{u}})_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right). \qquad (5)$$

The constitutive law for ice relates the deviatoric stress tensor τ to the strain-rate tensor $\dot{\varepsilon}_{\mathbf{u}}$ by the generalized Glen's flow law [*Nye*, 1957; *Paterson*, 1994]

$$\tau = 2\eta_{\mathbf{u}}\dot{\varepsilon}_{\mathbf{u}} \qquad (6)$$

with

$$\eta_{\mathbf{u}} = \frac{1}{2}A^{-1/n}\dot{\varepsilon}_e^{(1-n)/n}, \qquad (7)$$

where n is referred to as the power-law exponent, A as the deformation rate factor, $\eta_{\mathbf{u}}$ as the effective viscosity, and $\dot{\epsilon}_e$ as the effective strain-rate defined by

$$\dot{\epsilon}_e \;=\; \sqrt{\tfrac{1}{2}\dot{\varepsilon}_{\mathbf{u}} : \dot{\varepsilon}_{\mathbf{u}}}. \tag{8}$$

The deformation rate factor A depends on temperature and pressure, and possibly on other properties such as ice-crystal size and orientation and impurity content [*Paterson*, 1994]. In *Zhang et al.* [2011], it is assumed that A depends only on temperature and obeys an Arrhenius relation defined by

$$A = A(T) = a\exp\left(-Q/RT\right),$$

where a is an empirical flow constant often used as a tuning parameter, Q denotes the activation energy for creep, R denotes the universal gas constant, and T denotes the absolute temperature measured in degrees Kelvin. Because our study mainly focuses on building a numerical Stokes model for ice sheet dynamics, in this paper we assume an isothermal system in which A is a spatially uniform constant, i.e., we do not couple flow and temperature evolution by solving an equation for conservation of energy. More general cases will be treated in future work.

If the top surface of the ice sheet is allowed to evolve in time, then a prognostic equation describing the evolution of that free surface should also be included. The ice sheet domain Ω_t at a time t can be defined as

$$\begin{aligned}\Omega_t = \{(x,y,z) \mid z_b(x,y) \le z \le z_s(x,y,t),\\ (x,y)\in\Omega_H,\, t\in[0,t_{max}]\}\end{aligned} \tag{9}$$

where Ω_H denotes the horizontal extent of the ice sheet, $z_s(x,y,t)$ defines the top surface elevation, and $z_b(x,y)$ defines the bed elevation. Note that here $z_b(x,y)$ is assumed to be fixed with respect to the time, i.e., neither isostatic rebound or erosion is considered. We denote the top surface as Γ_s and the bottom surface as Γ_b. In general, $z_b(x,y) \neq z_s(x,y,t)$ along the boundary of Ω_H so that the ice sheet may also have a lateral boundary Γ_ℓ. The motion of the free surface is governed by the kinematic relation

$$\begin{aligned}\frac{\partial z_s}{\partial t} + u_x(x,y,z_s)\frac{\partial z_s}{\partial x} + u_y(x,y,z_s)\frac{\partial z_s}{\partial y}\\ -u_z(x,y,z_s) = b(x,y,z_s,t)\end{aligned} \tag{10}$$

on the top surface Γ_s, where $b(x,y,z_s,t)$ represents the surface mass balance. In general, $b \neq 0$ because of accumulation and ablation at the top surface.

2.2. Boundary conditions

At the top surface of the ice sheet, we impose the boundary condition

$$\sigma\cdot\mathbf{n} = -p_{atm}\cdot\mathbf{n} \qquad \text{on } \Gamma_s, \tag{11}$$

where $\mathbf{n}$ denotes the outer normal unit vector at the ice sheet boundary and p_{atm} the atmospheric pressure. Because atmospheric pressure is negligible relative to pressure within the ice column, we make the standard simplification that $p_{atm} = 0$.

Along the lateral boundary Γ_ℓ, we impose one of three types of boundary conditions; a condition such as (11), the zero velocity condition $\mathbf{u} = 0$, or periodic boundary conditions. Having this flexibility allows us to not only model realistic ice sheets, but also to apply our approach to benchmark examples for which the latter two non-physical boundary conditions are used. Note that in our current model and experiments we do not consider the case for which the lateral boundary is partially submerged in water such as occurs at an ice-ocean boundary.

The bottom surface of the ice sheet can be decomposed into two parts, $\Gamma_{b,fix}$ at which the ice sheet is fixed to the bottom bedrock and $\Gamma_{b,sld}$ at which it is allowed to slide. On the fixed part of the basal boundary, we impose the zero velocity boundary condition

$$\mathbf{u} = 0 \qquad \text{on } \Gamma_{b,fix}, \tag{12}$$

which includes the no-penetration condition $\mathbf{u}\cdot\mathbf{n} = 0$ and the no-sliding condition $\mathbf{u}\times\mathbf{n} = 0$. On the sliding part of the basal boundary, we impose the *Rayleigh friction* boundary condition

$$\mathbf{u}\cdot\mathbf{n} = 0 \quad \text{and} \quad \mathbf{n}\cdot\sigma\cdot\mathbf{t} = -\beta^2\mathbf{u}\cdot\mathbf{t} \qquad \text{on } \Gamma_{b,sld}. \tag{13}$$

We note that other type of friction laws such as Coulomb's law can be applied on $\Gamma_{b,sld}$; see [e.g., *Truffer et al.*, 2001; *Burstedde et al.*, 2009; *Schoof*, 2006, 2010]. Here, we consider the Rayleigh friction law largely to allow for comparison with benchmark experiments. The parameter β^2 denotes a given sliding coefficient and $\mathbf{t}$ denotes any unit vector tangential to the bottom surface. Note that the negative sign in (13) implies that the direction of the friction force is opposite to that of the velocity.

2.3. Variational formulation

Finite element discretizations are based on variational formulations of the partial differential equation system; in this section, we derive a variational formulation of the Stokes system (2) and (4) along with appropriate boundary conditions.

Let $L^2(\Omega_t)$ denote the space of square-integrable functions with respect to Ω_t. Let $\mathbf{H}^1(\Omega_t)$ denote the space of vector functions each of whose components belong to the space $H^1(\Omega_t)$ that consists of functions belonging to $L^2(\Omega_t)$ whose first derivatives also belong to $L^2(\Omega_t)$. Multiply (4) by a test function $\mathbf{v}\in\mathbf{H}_1(\Omega_t)$, then integrate the result over Ω_t. Integration by parts results in

$$\int_{\Omega_t}\tau : \nabla\mathbf{v}\, d\mathbf{x} - \int_{\Omega_t} p\nabla\cdot\mathbf{v}\, d\mathbf{x} - \int_{\Gamma}\mathbf{n}\cdot\sigma\cdot\mathbf{v}\, ds = \rho\int_{\Omega_t}\mathbf{g}\cdot\mathbf{v}\, d\mathbf{x}, \tag{14}$$

where $\Gamma = \Gamma_s\cup\Gamma_b\cup\Gamma_\ell$ and $\tau : \nabla\mathbf{v}$ denotes the sum of the element-wise products of the tensors τ and $\nabla\mathbf{v}$. Note that because of the symmetry of the stress tensor τ, we have $\tau_{ij}v_{i,j} = \tau_{ji}v_{j,i} = \tau_{ij}v_{j,i}$, where we use the repeated index summation convention and $v_{i,j} = \partial v_i/\partial x_j$. Then, $\tau_{ij}v_{i,j} = \frac{1}{2}\tau_{ij}(v_{i,j}+v_{j,i}) = \tau_{ij}(\dot{\varepsilon}_{\mathbf{v}})_{ij} = 2\eta_{\mathbf{u}}(\dot{\varepsilon}_{\mathbf{u}})_{ij}(\dot{\varepsilon}_{\mathbf{v}})_{ij}$, where for the last equality we use (6). Thus, we have

$$\int_{\Omega_t}\tau : \nabla\mathbf{v}\, d\mathbf{x} = \int_{\Omega_t} 2\eta_{\mathbf{u}}\dot{\varepsilon}_{\mathbf{u}} : \dot{\varepsilon}_{\mathbf{v}}\, d\mathbf{x}. \tag{15}$$

From (11) with $p_{atm} = 0$, we have

$$\int_{\Gamma_s}\mathbf{n}\cdot\sigma\cdot\mathbf{v}\, ds = 0. \tag{16}$$

If a condition such as (11) is imposed on Γ_ℓ, we also have (with $p_{atm} = 0$)

$$\int_{\Gamma_\ell}\mathbf{n}\cdot\sigma\cdot\mathbf{v}\, d\mathbf{x} = 0. \tag{17}$$

If a periodic boundary condition is applied on Γ_ℓ, we require that the test functions $\mathbf{v}$ be periodic and then (17) holds as well. If the zero velocity boundary condition is applied on Γ_ℓ, we require $\mathbf{v} = \mathbf{0}$ on Γ_ℓ so that (17) still holds.

On the no-sliding part $\Gamma_{b,fix}$ of the basal boundary, we require that the test function satisfy $\mathbf{v} = \mathbf{0}$ so that

$$\int_{\Gamma_{b,fix}} \mathbf{n} \cdot \sigma \cdot \mathbf{v}\, ds = 0. \tag{18}$$

On the sliding part $\Gamma_{b,sld}$ of the basal boundary, because $\mathbf{u} \cdot \mathbf{n} = 0$, we require the test function to satisfy $\mathbf{v} \cdot \mathbf{n} = 0$. Together with the friction law in (13), we then have that

$$\int_{\Gamma_{b,sld}} \mathbf{n} \cdot \sigma \cdot \mathbf{v}\, ds = -\int_{\Gamma_{b,sld}} \beta^2 \mathbf{u} \cdot \mathbf{v}\, ds. \tag{19}$$

Let

$$\widetilde{\mathbf{H}}(\Omega_t) = \{\mathbf{u} \in \mathbf{H}^1(\Omega_t) \mid \mathbf{u}|_{\Gamma_l \cup \Gamma_{b,fix}} = 0,\ (\mathbf{u} \cdot \mathbf{n})|_{\Gamma_{b,sld}} = 0\}.$$

Note that functions in $\widetilde{\mathbf{H}}(\Omega_t)$ satisfy homogeneous boundary conditions on the indicated parts of the boundary. Substituting (15)–(19) into (14), we obtain the weak formulation of the nonlinear Stokes model (2) and (4) along with the boundary conditions: seek $\mathbf{u} \in \widetilde{\mathbf{H}}(\Omega_t)$ and $p \in L^2(\Omega_t)$ such that

$$\begin{cases} \int_{\Omega_t} 2\eta_{\mathbf{u}} \dot\varepsilon_{\mathbf{u}} : \dot\varepsilon_{\mathbf{v}}\, d\mathbf{x} + \int_{\Gamma_{b,sld}} \beta^2 \mathbf{u} \cdot \mathbf{v}\, ds \\ \qquad - \int_{\Omega_t} p \nabla \cdot \mathbf{v}\, d\mathbf{x} = \rho \int_{\Omega_t} \mathbf{g} \cdot \mathbf{v}\, d\mathbf{x}, \\ -\int_{\Omega_t} q \nabla \cdot \mathbf{u}\, d\mathbf{x} = 0, \end{cases} \tag{20}$$

for all $\mathbf{v} \in \widetilde{\mathbf{H}}(\Omega_t)$ and $q \in L^2(\Omega_t)$. Specifically, (20) corresponds to a zero velocity lateral boundary condition. If a lateral boundary condition similar to (11) with $p_{atm} = 0$ is imposed instead, then the requirement that the functions in $\widetilde{\mathbf{H}}(\Omega_t)$ vanish on Γ_ℓ is removed. If periodic lateral boundary conditions are imposed, then that requirement is replaced by the requirement that the functions in $\widetilde{\mathbf{H}}(\Omega_t)$ be periodic.

If the free-surface equation (10) is used to evolve the ice sheet, then its weak form is to find $z_s \in H^1(\Omega_H)$, such that

$$\int_{\Gamma_s} \frac{\partial z_s}{\partial t} \psi\, ds = \int_{\Gamma_s} \left(u_z - u_x \frac{\partial z_s}{\partial x} - u_y \frac{\partial z_s}{\partial y} + b \right) \psi\, ds,$$

for all $\psi \in H^1(\Omega_H)$. Temporal discretization is effected by first dividing the time interval $[0, t_{max}]$ into K subintervals $\{[t_{k-1}, t_k]\}_{k=1}^K$, where $t_k = k\Delta t$ and $\Delta t = t_{max}/K$. Then, a semi-discrete (in time) variational problem using the forward Euler scheme is to seek z_s^{k+1} on Γ_{s,t_k} such that

$$\begin{aligned} \int_{\Gamma_{s,t_k}} z_s^{k+1} \psi\, ds = \int_{\Gamma_{s,t_k}} z_s^k \psi\, ds + \\ \int_{\Gamma_{s,t_k}} \Delta t \left(u_z^k - u_x^k \frac{\partial z_s^k}{\partial x} - u_y^k \frac{\partial z_s^k}{\partial y} + b^k \right) \psi\, ds, \end{aligned} \tag{21}$$

for $k = 0, 1, \cdots, K-1$ where z_s^k denotes the top surface height at time t_k, b^k the surface mass balance at t_k, Γ_{s,t_k} the top surface of the ice sheet at t_k, and z_s^0 is given as an initial condition.

3. High-order accurate finite element Stokes ice sheet model

In this section, we provide a detailed description of the high-order accurate finite element Stokes ice sheet model. In Section 3.1, we discuss the finite element discretization, leading to the nonlinear system of discrete equations that have to be solved to determine approximate solutions. Section 3.1 also includes a discussion of the Picard linearization of the nonlinear discrete system as well as other details about the discretization process. Then, in Section 3.2, we provide details about the efficient parallel algorithm we use to solve the linear algebraic systems resulting from the Picard linearization. The finite element discretization and parallel linear system solvers together form our numerical Stokes ice sheet model.

3.1. High-order accurate finite element discretization

3.1.1. Tetrahedral grid generation

The geometry of ice sheets such as Greenland and Antarctica is highly anisotropic, with representative horizontal to vertical length scales having ratios from 100 : 1 to 1000 : 1. Moreover, in large portions of the ice sheets, variations in descriptive variables in horizontal directions are much smaller than those in the vertical direction. As a result, the use of three-dimensional isotropic meshing of the highly anisotropic geometry would result in a huge number of grid points and therefore, many more degrees of freedom then is necessary or viable. Therefore, anisotropy of the computational grid is called for when obtaining high-order accurate solutions with relatively few degrees of freedom. Except near the ice sheet boundaries and in areas of concentrated flow (e.g., ice streams and outlet glaciers), solutions are expected to vary much more slowly in the horizontal directions than in the vertical direction. This allows for relatively large grid spacing in the horizontal direction over large regions of the ice sheets where the solution is mostly slowly varying. Also note that the high aspect ratio of ice sheets cannot be avoided merely through rescaling of the vertical coordinate; doing this can indeed improve the domain aspect ratio, but it also changes the coefficients in the partial differential equations so that the physical aspect ratio will still appear in detrimental ways in the discretized systems.

Due to the high aspect ratio of the ice sheet, which results in the need for high-aspect ratio anisotropic grids, some special treatment during the generation of tetrahedral grids is needed to avoid low-quality elements. Our approach is to first generate a high-quality, two-dimensional triangular grid for the horizontal extent of the ice sheet Ω_H, see *Zhang et al.* [2011] and the references cited therein; we denote the two-dimensional mesh by $\mathcal{Q}_h$. The two-dimensional grid is then transformed to a three-dimensional surface triangular grid for the top surface of the ice sheet by adding the z-coordinate obtained from the topography of the bedrock to which the ice thickness is added. We then produce a fully three-dimensional, layered, prismatic mesh of the ice sheet domain Ω_t by subdividing the thickness at each horizontal grid point into equal increments so that the number of vertical increments is the same at each grid point of the horizontal mesh. Finally, we obtain a tetrahedral grid of the ice sheet by decomposing each of the prismatic elements into three tetrahedra.

In general, the top surface of the ice sheet evolves in time, after the computation of the top surface elevation z_s; see Section 3.1.5. To avoid grid distortion, the grid points should be redistributed according to the change in the ice thickness. Linear or nonlinear elasticity analogs could be used to deform the mesh. Here, for simplicity, a linear mapping is applied to determine the z-coordinates of the mesh vertices at each time step for which vertical regridding takes place. The governing equation is given by

$$\begin{aligned} \frac{\partial Z}{\partial z} &= 1 && \text{on } \Omega_t, \\ Z &= z_s && \text{on } \Gamma_s, \\ Z &= z_b && \text{on } \Gamma_b, \end{aligned} \tag{22}$$

where Z denotes the z-coordinate of the mesh vertices. If the initial mesh is uniform in the z direction at each horizontal grid point, this linear mapping ensures that subsequent meshes are likewise uniform in the z direction.

Because the ice sheet margin can advance or recede, mechanisms should also be introduced to deal with lateral changes in thickness and domain extent. The examples considered in this paper do not experience such thickness changes, so our current Stokes model has not incorporated such mechanism. However, when necessary, e.g., when considering realistic ice sheets and and when coupling ice sheet dynamics with ice sheet energetics, such changes are allowed to occur. They can be dealt with by, e.g., allowing for the deletion or addition of elements in the vertical direction when the ice thickness becomes too small or too large, respectively. Such mechanisms will be incorporated into our thermo-mechanical model in the future.

The examples used in our numerical studies in Section 4 involve periodic boundary conditions in at least some directions. It is useful for grids to match at opposite ends of the computational domain in directions for which periodic boundary conditions are applied. Otherwise, enforcing the periodicity, including the periodicity of test and trial functions (see Section 2.3), would require interpolations that can introduce significant errors. In our numerical experiments, because of the simple geometries and solution features, quasi-uniform grids are appropriate; such grids also facilitate easy matching of grids at periodic boundaries. In ongoing work, the codes we are developing for the Greenland and Antarctic ice sheets use highly nonuniform grids and in these cases periodicity is not an issue.

3.1.2. Finite element discretization of the Stokes system

Let $\mathcal{T}_h$ denote a tetrahedral triangulation of the ice sheet domain Ω_t generated as discussed in Section 3.1.1. Here, h is a measure of the spatial grid size, e.g., the maximum diameter of the circumspheres of all tetrahedral elements. The finite element space $P_{1,h}(\mathcal{T}_h)$ used for approximating the pressure consists of functions that, within each tetrahedral element, are linear polynomials, e.g., in x, y, z space, they are functions of the form $a_0 + a_1x + a_2y + a_3z$ for constants a_i, $i = 0, \ldots, 3$. Such functions are uniquely determined by their values at the four vertices of a tetrahedral element. For approximation of the components of velocity, we instead use a higher-order finite element space, $P_{2,h}(\mathcal{T}_h)$, that consists of quadratic polynomials inside each of the tetrahedra, e.g., in x, y, z space, they are functions of the form $b_0+b_1x+b_2y+b_3z+b_4x^2+b_5y^2+b_6z^2+b_7xy+b_8yz+b_9zx$ for constants b_i, $i = 0, \ldots, 9$. These functions can be uniquely determined by their values at the four vertices and the six edge midpoints of a tetrahedron. See Figure 1 for an illustration. In addition, the functions in both $P_{1,h}$ and $P_{2,h}$ are required to be continuous across element faces. This pair of elements for discretizing the Stokes equation is referred to as the Taylor-Hood element pair and satisfies the LBB stability condition (or inf-sup condition) [*Gunzburger*, 1989] required for stable approximations. We then define the constrained space (in case the lateral boundary condition is a zero velocity condition)

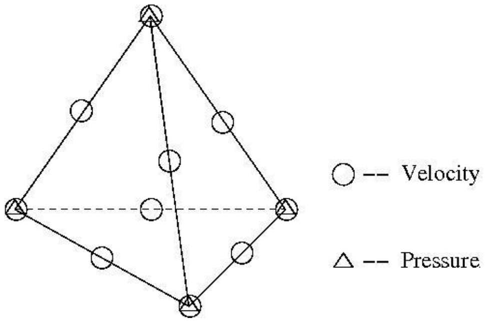

Figure 1. The Taylor-Hood (P2-P1) element pair.

$$\widetilde{\mathbf{P}}_{2,h}(\mathcal{T}_h) = \{\mathbf{u}_h \in (P_{2,h}(\mathcal{T}_h))^3 \mid \mathbf{u}_h|_{\Gamma_l \cup \Gamma_{b,fix}} = 0, \quad (\mathbf{u}_h \cdot \mathbf{n})|_{\Gamma_{b,sld}} = 0\}. \tag{23}$$

Thus, given Ω_t and $\mathcal{T}_h$, we seek functions $\mathbf{u}_h \in \widetilde{\mathbf{P}}_{2,h}(\mathcal{T}_h)$ and $p_h \in P_{1,h}(\mathcal{T}_h)$ such that

$$\begin{cases} \displaystyle\int_{\Omega_t} 2\eta_{\mathbf{u_h}} \dot{\varepsilon}_{\mathbf{u}_h} : \dot{\varepsilon}_{\mathbf{v}_h} \, d\mathbf{x} + \int_{\Gamma_{b,sld}} \beta^2 \mathbf{u}_h \cdot \mathbf{v}_h \, ds - \\ \qquad\qquad \displaystyle\int_{\Omega_t} p_h \nabla \cdot \mathbf{v}_h \, d\mathbf{x} = \rho \int_{\Omega_t} \mathbf{g} \cdot \mathbf{v}_h \, d\mathbf{x}, \\ \displaystyle -\int_{\Omega_t} q_h \nabla \cdot \mathbf{u}_h \, d\mathbf{x} = 0, \end{cases} \tag{24}$$

for all $\mathbf{v}_h \in \widetilde{\mathbf{P}}_{2,h}(\mathcal{T}_h)$ and $q_h \in P_{1,h}(\mathcal{T}_h)$. Due to the dependence of η on the approximate velocity $\mathbf{u}_h$, (24) is a nonlinear system of equations for the approximate velocity and pressure $\mathbf{u}_h$ and p_h, respectively.

It is known that the error of the Taylor-Hood approximation for the linear Stokes equations (i.e. for Newtonian fluids) is of third-order accuracy in the velocity $\mathbf{u}$ (i.e., $O(h^3)$) and of second-order accuracy in the pressure p (i.e., $O(h^2)$) [*Bercovier and Pironneau*, 1979]. We expect the orders of approximations to be similar in the nonlinear setting; this is partially verified by the numerical example of Section 4.1.

There are several reasons for not using even higher-order elements, including the fact that solutions may not have sufficient smoothness to take advantage of any additional accuracy. However, the most important reason for not going to even higher-order elements is that for realistic ice sheet geometries, the boundary is given discretely, i.e., as a set of points, and that boundary can be extremely irregular and made up of very short segments. In the end, the lack of exact knowledge of the boundary geometry is a controlling factor in determining solution accuracy so that higher-order elements are not necessarily beneficial.

3.1.3. Picard linearization

We use a simple and direct Picard-type iterative algorithm to solve the nonlinear system (24). More sophisticated linearization algorithms that promise faster convergence (e.g., Newton's method or quasi-Newton methods) that require the evaluation of Jacobian or approximate Jacobian matrices are also of interest. Compared to the simple Picard method, the latter types of approaches often require a better initial iterate for convergence. That is, the initial iterate must be close to the desired solution of the nonlinear system. As a result, the application of those methods is often preceded by a few steps of a Picard-type iteration to ensure that when one switches to the more sophisticated and faster converging method, convergence will result.

The Picard iteration simply lags the velocity-dependent viscosity $\eta_{\mathbf{u}_h}$ in (24), i.e., at any iteration of the Picard method, $\eta_{\mathbf{u}_h}$ is evaluated using the approximate velocity solution obtained at the previous step. Thus, starting with an initial guess $\mathbf{u}_h^{(0)}$ for the velocity approximation (which is often taken to be $\mathbf{u}_h^{(0)} = \mathbf{0}$ and in this case $\eta_{\mathbf{u}_h^0}$ is set to be certain positive constant), we have that, for $j = 1, 2, \ldots$, $\mathbf{u}_h^{(j)}$ and $p_h^{(j)}$ are determined by solving the *linear* problem

$$\begin{cases} \displaystyle\int_{\Omega_t} 2\eta_{\mathbf{u}_h^{(j-1)}} \dot{\varepsilon}_{\mathbf{u}_h^{(j)}} : \dot{\varepsilon}_{\mathbf{v}_h} \, d\mathbf{x} + \int_{\Gamma_{b,sld}} \beta^2 \mathbf{u}_h^{(j)} \cdot \mathbf{v}_h \, ds - \\ \qquad\qquad \displaystyle\int_{\Omega_t} p_h^{(j)} \nabla \cdot \mathbf{v}_h \, d\mathbf{x} = \rho \int_{\Omega_t} \mathbf{g} \cdot \mathbf{v}_h \, d\mathbf{x}, \\ \displaystyle -\int_{\Omega_t} q_h \nabla \cdot \mathbf{u}_h^{(j)} \, d\mathbf{x} = 0. \end{cases} \tag{25}$$

Finally, we set $\mathbf{u}_h = \mathbf{u}_h^{(j)}$ when satisfactory convergence of the Picard iterates is achieved in the sense that the residual relative to the approximate solution is smaller than a prescribed tolerance. Using a simple heuristic asymptotic analysis, we expect that the Picard iteration is linearly convergent with an contraction constant $\frac{n-1}{n}$, where n denotes the exponent in the Glen's law.

The Picard iteration produces, at each step, the linear finite element problem (25) which is equivalent to a symmetric saddle-point problem of the type

$$\begin{pmatrix} F & B^T \\ B & \mathbf{0} \end{pmatrix} \begin{pmatrix} \vec{\mathbf{u}} \\ \vec{p} \end{pmatrix} = \begin{pmatrix} \vec{\mathbf{r}} \\ 0 \end{pmatrix}, \tag{26}$$

where $\vec{\mathbf{u}}$ and $\vec{p}$ denote the vectors of velocity and pressure degrees of freedom, respectively. Thus, once (25), or equivalently, (26) is assembled, there only remains the need to define an efficient solver for that linear system. The linear solver strategy we use is discussed in Section 3.2.

3.1.4. Implementation of the no-penetration condition on the sliding boundary

There remains the important problem of how to implement the sliding boundary condition (13), specifically how the no-penetration condition $\mathbf{u} \cdot \mathbf{n} = 0$ is implemented along boundaries that are not aligned with the coordinate axes. The difficulty arises because such a condition involves, in general, a linear combination of all three components of the velocity. Our approach is based on the well-known technique from *Nastran Documentation* [2009]. The main idea to use a rotated coordinate system at each velocity node on the sliding boundary $\Gamma_{b,sld}$. The coordinate system is chosen to have one of the coordinate directions orthogonal to the boundary $\Gamma_{b,sld}$ at the node and the other two tangent to the boundary. Then, the no-penetration condition is easy to apply on the normal direction in this rotated frame, as we now explain.

Assume that the number of velocity degrees of freedom (DOF) is $3M$, where M denotes the number of grid points for velocity including the mid-sides of the tetrahedra. Then, we rewrite the velocity vector $\vec{\mathbf{u}}$ in (26) in the form

$$\vec{\mathbf{u}} = \begin{pmatrix} \mathbf{u}_1 \\ \vdots \\ \mathbf{u}_M \end{pmatrix}$$

with $\mathbf{u}_k = (u_{k,x}, u_{k,y}, u_{k,z})^T$. The no-penetration constraint on $\Gamma_{b,sld}$ is given by

$$\mathbf{n}_k \cdot \mathbf{u}_k = 0 \qquad \text{for all } k \in \sigma_{b,sld}, \tag{27}$$

where $\sigma_{b,sld}$ denotes the set of indices of the DOF of $\mathbf{u}$ that lie on the sliding boundary $\Gamma_{b,sld}$ and $\mathbf{n}_k$ denotes the outer normal of the surface at the velocity grid point corresponding to corresponding to $\mathbf{u}_k$.

For each DOF $k \in \sigma_{b,sld}$ on the sliding boundary, a local coordinate system is built aligned with the surface normal vector $\mathbf{n}_k$ and two orthonormal surface tangential vectors $\mathbf{t}_k^1$ and $\mathbf{t}_k^2$. Define the 3×3 transformation matrix

$$T_k = (\mathbf{n}_k, \mathbf{t}_k^1, \mathbf{t}_k^2).$$

On the other hand, for the DOF that do not lie on the sliding boundary, define the corresponding transformation matrix to be the 3×3 identity matrix I, i.e., $T_k = I$ if $k \notin \sigma_{b,sld}$. Then, we obtain the global transformation matrix for all k

$$T = \begin{pmatrix} T_1 & & \\ & \ddots & \\ & & T_M \end{pmatrix}.$$

Note that $T^T T = I$. Inserting T into the linear system (26), we have

$$\begin{pmatrix} \widetilde{F} & \widetilde{B}^T \\ \widetilde{B} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \tilde{\vec{\mathbf{u}}} \\ \vec{p} \end{pmatrix} = \begin{pmatrix} \tilde{\vec{\mathbf{r}}} \\ 0 \end{pmatrix} \tag{28}$$

where

$$\widetilde{F} = TFT^T, \qquad \widetilde{B} = BT^T, \qquad \tilde{\vec{\mathbf{u}}} = T\vec{\mathbf{u}}, \qquad \tilde{\vec{\mathbf{r}}} = T\vec{\mathbf{r}}.$$

Now, let us use $\tilde{\vec{\mathbf{u}}}$ as unknowns. Note that

$$\tilde{\vec{\mathbf{u}}}_k = (\mathbf{n}_k \cdot \mathbf{u}_k, \mathbf{t}_k^1 \cdot \mathbf{u}_k,\ \mathbf{t}_k^2 \cdot \mathbf{u}_k)^T = (0, \mathbf{t}_k^1 \cdot \mathbf{u}_k,\ \mathbf{t}_k^2 \cdot \mathbf{u}_k)^T$$

for any $k \in \sigma_{b,sld}$. Thus, to apply the no-penetration condition, we simply set, for $k \in \sigma_{b,sld}$, the first component of each $(\tilde{\vec{\mathbf{u}}})_k$ to zero. Then, after solving (28) for $\tilde{\vec{\mathbf{u}}}$, we recover the solution of (26) by simply setting $\vec{\mathbf{u}} = T^T(\tilde{\vec{\mathbf{u}}})$.

3.1.5. Finite element discretization for the top surface evolution

For the approximation of the top surface $z_s(x, y, t)$ during evolution, we use the linear finite element space $P_{1,h}(\mathcal{Q}_h)$ consisting of continuous piecewise linear polynomials defined with respect to the two-dimensional triangular triangulation $\mathcal{Q}_h$ of the horizontal extent of the ice sheet, i.e., of Ω_H. Note that we assume that Ω_H is fixed. Then, the fully discretized (in time and space) free-surface equation can be described as follows: for $k = 0, 1, \ldots, K-1$, given $\mathbf{u}_h^k$, we solve $z_{s,h}^{k+1} \in P_{1,h}(\mathcal{Q}_h)$

$$\begin{aligned} \int_{\Gamma_{s,t_k}} z_{s,h}^{k+1} \psi_h \, ds = \int_{\Gamma_{s,t_k}} & z_{s,h}^{k} \psi_h \, ds + \\ \int_{\Gamma_{s,t_k}} & \Delta t \left(u_{h,z}^k - u_{h,x}^k \frac{\partial z_{s,h}^k}{\partial x} - u_{h,y}^k \frac{\partial z_{s,h}^k}{\partial y} + b^k \right) \psi_h \, ds \end{aligned} \tag{29}$$

for all $\psi_h \in P_{1,h}(\mathcal{Q}_h)$. Thus, once $\mathbf{u}_h^k$ has been determined, we can easily update the elevation of the top surface $z_{s,h}^{k+1}$ and thus determine the new domain $\Omega_{t_{k+1}}$. Note that at each time step, after the computation of the surface elevation, the grid points are redistributed in the vertical direction using the remeshing procedure discussed in Section 3.1.1.

3.2. Iterative linear solvers, preconditioners, and parallelization

The large, sparse linear systems such as (28) arising from a finite element discretization may have millions of unknowns in order to obtain high-resolution in the numerical simulations. Solving such large-scale systems is a challenging task due to the high demand on computing power and memory. Iterative solution techniques for massive linear systems based on Krylov subspace methods (such as GMRES and conjugate gradient methods) as well as preconditioning techniques (such as block, multigrid, and incomplete LU factorization preconditioners [*Li and Demmel*, 2003]) are commonly used because the Krylov subspace iteration methods require only matrix-vector products. Here, we discuss the parallel preconditioned iterative method we use to solve the linear system (28) produced at each step of the Picard iteration (25) that linearizes the nonlinear discretized system (24). Our choice of preconditioned linear solver strategy is motivated by the saddle-point and possibly very ill-conditioned nature of the coefficient matrix in (28) [*Elman et al.*, 2005].

3.2.1. Block preconditioner

Consider the block factorization of the coefficient matrix in (28) given by

$$\begin{pmatrix} \widetilde{F} & \widetilde{B}^T \\ \widetilde{B} & \mathbf{0} \end{pmatrix} = \begin{pmatrix} I & 0 \\ \widetilde{B}\widetilde{F}^{-1} & I \end{pmatrix} \begin{pmatrix} \widetilde{F} & \widetilde{B}^T \\ 0 & -S \end{pmatrix}, \tag{30}$$

where $S = \widetilde{B}\widetilde{F}^{-1}\widetilde{B}^T$ denotes the Schur complement. Ideally, one would use the right factor

$$\begin{pmatrix} \widetilde{F} & \widetilde{B}^T \\ 0 & -S \end{pmatrix} \tag{31}$$

in (30) as a "perfect" right block preconditioner for the linear system (28). In fact, GMRES with this preconditioner converges in at most two iterations. However, the application of (31) as a preconditioner in an iterative method such as GMRES requires the multiplication of a vector by the inverse of that matrix, i.e., by the matrix

$$\begin{pmatrix} \widetilde{F}^{-1} & \widetilde{F}^{-1}\widetilde{B}^T S^{-1} \\ 0 & -S^{-1} \end{pmatrix} = \begin{pmatrix} \widetilde{F}^{-1} & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} I & -\widetilde{B}^T \\ 0 & I \end{pmatrix} \begin{pmatrix} I & 0 \\ 0 & -S^{-1} \end{pmatrix}. \tag{32}$$

For the sake of efficiency, we want to avoid having to deal with the application of the matrix $S^{-1} = (\widetilde{B}\widetilde{F}^{-1}\widetilde{B}^T)^{-1}$ that, among other difficulties, would involve an application of $\widetilde{F}^{-1}$ [*Silvester et al.*, 2001]. Fortunately, one can replace S in (31) by any matrix that is spectrally equivalent to S and not affect the performance of the the block preconditioner (31). To this end, we replace S by the weighted mass matrix M_η, where $(M_\eta)_{i,j} = \int_{\Omega_t} (\eta_{\mathbf{u}_h^{(j-1)}})^{-1}\phi_i\phi_j d\mathbf{x}$ and ϕ_i denotes a pressure basis function. Note that M_η is determined using the previous Picard iterate for the velocity and is fixed throughout the solution process for (30) that determines the new Picard iterate. The matrix M_η is easier to apply compared to S and is spectrally equivalent to S [*Olshanskii and Reusken*, 2006; *Olshanskii et al.*, 2006]. Thus, instead of (32), we use

$$\begin{pmatrix} \widetilde{F}^{-1} & \widetilde{F}^{-1}\widetilde{B}^T M_\eta^{-1} \\ 0 & -M_\eta^{-1} \end{pmatrix} = \begin{pmatrix} \widetilde{F}^{-1} & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} I & -\widetilde{B}^T \\ 0 & I \end{pmatrix} \begin{pmatrix} I & 0 \\ 0 & -M_\eta^{-1} \end{pmatrix} \tag{33}$$

as an approximation to the inverse of the preconditioner (31) so that, instead of (28), we solve the preconditioned linear system

$$\begin{pmatrix} \widetilde{F} & \widetilde{B}^T \\ \widetilde{B} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \widetilde{F}^{-1} & \widetilde{F}^{-1}\widetilde{B}^T M_\eta^{-1} \\ \mathbf{0} & -M_\eta^{-1} \end{pmatrix} \begin{pmatrix} \vec{\widetilde{\mathbf{v}}} \\ \vec{q} \end{pmatrix} = \begin{pmatrix} \vec{\widetilde{\mathbf{r}}} \\ 0 \end{pmatrix} \tag{34}$$

for $\vec{\widetilde{\mathbf{v}}}$ and $\vec{q}$, where

$$\begin{pmatrix} \vec{\widetilde{\mathbf{u}}} \\ \vec{p} \end{pmatrix} = \begin{pmatrix} \widetilde{F}^{-1} & \widetilde{F}^{-1}\widetilde{B}^T M_\eta^{-1} \\ \mathbf{0} & -M_\eta^{-1} \end{pmatrix} \begin{pmatrix} \vec{\widetilde{\mathbf{v}}} \\ \vec{q} \end{pmatrix}. \tag{35}$$

We apply a GMRES iteration to solve the linear system (34). The application of $\widetilde{F}^{-1}$ dominates the cost of that iteration. The block-preconditioning strategy just described is very effective when the subproblem involving $\widetilde{F}^{-1}$ is solved exactly. Of course, because of the large number of degrees of freedom involved, we can only solve that subproblem iteratively, in which case the preconditioner is highly sensitive to the residual of the $\widetilde{F}^{-1}$ solution. The approach we take for obtaining this solution is discussed in Section 3.2.2. Since M_η is a symmetric, positive definite, very well conditioned matrix the subproblem involving M_η^{-1} is very easy to solve; we use a standard preconditioned conjugate gradient matrix for this purpose. Note that we solve the block subproblems involving $\widetilde{F}^{-1}$ (see Section 3.2.2) and M_η^{-1} using iterative methods so that those subproblems are solved inexactly; as a result, the approximations of $\widetilde{F}^{-1}$ and M_η^{-1} at each GMRES iteration change as does their application to vectors. For this reason, we use the FGMRES variant [*Frayssé et al.*, 2009; *Saad*, 1993] because it is specifically designed to handle matrices that change at each iteration.

3.2.2. Solving the velocity subproblem

Each iteration of the FGMRES method requires the application of $\widetilde{F}^{-1}$ to a velocity vector iterate; for the sake of efficiency, we would also like to use an iterative method for this application. A solver for the subproblem involving $\widetilde{F}^{-1}$ should at the least properly handle the following three obstacles:

1. *anisotropy* – the computing grid could have very thin shaped elements, e.g., with aspect ratio up to 100;
2. *highly variable viscosity* – the nonlinear viscosity can vary by a factor of 1000 across the computational domain;
3. *symmetric gradient* – the governing equation uses the symmetric gradient $\nabla\mathbf{u} + \nabla\mathbf{u}^T$ so that the subproblem is more like a linear elasticity problem and not a vector Poisson problem.

The GMRES method with an algebraic multigrid (AMG) preconditioning [*Briggs et al.*, 2000] is a reasonable solver for $\widetilde{F}^{-1}$ because it can handle highly variable coefficients and anisotropy; furthermore, it is a very efficient solver for linear elasticity problems if a proper strategy is used. Geometric multigrid methods could probably also play a role here because techniques such as semi-coarsening and line smoothing are quite effective for anisotropic problems. However, in the case of ice sheet modeling, the high variability of the viscosity is the main difficulty that has to be overcome by, e.g., using operator-dependent interpolation and Galerkin coarse-grid operators, and these techniques all fall into the AMG category. As described above, in our parallel solvers, we use an AMG-preconditioned GMRES method as one of the core steps in the solution of linear systems.

3.2.3. Parallelization

Parallel computing algorithms are often based on a divide and conquer strategy to solve large-scale problems. We adopt a domain decomposition method (DDM) [*Douglas*, 1996; *Valli*, 1999] for construction of the coefficient matrix and the local preconditioning on distributed computer processors. The finite element meshes are first partitioned into a number of submeshes, one for each processor to be used in the parallel calculation. This results in dividing the computing domain Ω_{t_k} at the time step t_k into interconnected subdomains. By doing so, the original large-scale computing problem is decomposed into a group of relatively simpler and smaller problems on different processors. Our grid partitioning is done only in the horizontal directions using "METIS," a family of multilevel mesh partitioning algorithms [*Karypis*, 2003]; see Figure 2 for an illustration.

Based on this partitioning approach, we parallelize all steps in our solution algorithm for (28), including all matrix-vector multiplies encountered in the two GMRES type iterations that are part of that algorithm. As described in Section 3.2.2, we use an AMG-preconditioned GMRES method as the core steps in the solution of (28). The parallel AMG solver *BoomerAMG* from the *hypre* package [*Falgout and Yang*, 2002] is used in our parallel implementation. *BoomerAMG* has a great flexibility for choosing between various parallel coarsening strategies and different interpolation operators. The AMG settings greatly affect the efficiency of

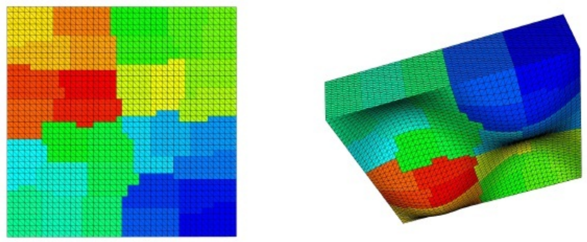

Figure 2. Partition of a sample tetrahedral grid into 32 subgrids. Left: two-dimensional view from the top; right: three-dimensional view.

the solver; choices for the AMG settings for different problems are discussed in Section 5.

Message Passing Interface (MPI) is taken as the parallel environment. As discussed above, in our implementation, we use the GMRES type methods in two places together with block and AMG preconditioning techniques for the solution of (34); in particular, the popular software package PETSc [*Balay et al.*, 2004] is employed in our parallel implementation due to its reliability and robustness. Some detailed discussions are given in Section 5. We also note that, for the above horizontal partitioning, parallel coarsening using AMG will have fewer difficulties in handling inter-processor boundary and data communication.

3.3. Summary of solver strategy for the nonlinear discrete system

There are several iterations involved in the solution process:

- the outermost Picard iteration for the nonlinear discrete system that produces a sequence of linear systems; see Section 3.1.3;
- each linear system is solved by an *outer* block-preconditioned FGMRES iteration; see Section 3.2.1;
- in turn, each outer FGMRES iterate requires a second GMRES iteration to solve the $\widetilde{F}^{-1}$ subproblem; see Section 3.2.2;
- the inner GMRES iteration is preconditioned by AMG; see Sections 3.2.2 and 3.2.3;
- all steps are parallelized as described in Section 3.2.3.

The stopping criterion for the outer Picard iteration is that the relative residual is less than 10^{-4}; for the outer FGMRES iteration, the stopping criterion is that the relative residual is less than 10^{-6}. Because the time-consuming AMG preconditioner is applied in the inner GMRES iteration for approximately solving the subproblem, we should use as few iterations as possible. Thus, for the inner AMG-preconditioned GMRES iteration we simply require the relative residual to reduce to roughly 10^{-1} (see Section 5 for more specific settings). These choices are made to ensure the accuracy and efficiency of the linear system solve.

4. Numerical tests

In this section, we demonstrate the high-order accuracy of our parallel finite element Stokes ice sheet model using an example with a known exact solution. We also provide results of some validations through the established ISMIP-HOM benchmark problems [*Pattyn et al.*, 2008].

4.1. Convergence tests

The convergence tests are performed using an isothermal Stokes ice sheet model with a manufactured exact solution; in particular, we use a two-dimensional steady-state analytical solution proposed in *Sargent and Fastook* [2010] and then extend it to three dimensions to illustrate the convergence rates of our high-order (P2-P1) finite element solver, and to compare its efficiency with the less accurate linear (P1-P1) finite element method. We note that our example can be regarded as a correction to the three-dimensional analytical solution constructed in *Sargent and Fastook* [2010], in which the solution to a first-order partial differential equation for the velocity is incorrect. The test considers ice flow over a domain of $0 \leq x \leq 80\,\mathrm{km}$, $s(x) \leq z \leq b(x)$, where $s(x)$ and $b(x)$ define the top and bedrock surfaces of the ice sheet, respectively. Following [*Sargent and Fastook*, 2010], we choose the following typical values: $Z = 1\,\mathrm{km}$ as the mean thickness of the ice sheet, $L = 80\,\mathrm{km}$ as the length of the ice sheet, U as the velocity in horizontal direction, W as the velocity in the vertical direction, P as the mean pressure, and $\delta = \frac{Z}{L}$ as the (inverse) aspect ratio parameter. Let the following relations hold: $\frac{1}{2}A^{-\frac{1}{n}}(\frac{U}{L})^{\frac{1}{n}} = \rho g Z = P$, and $\frac{WL}{UZ} = 1$. Then, the nondimensional variables (denoted with hats) are introduced as:

$$z = Z\widehat{z}, \quad s = Z\widehat{s}, \quad b = Z\widehat{b}, \quad x = L\widehat{x},$$

$$\mu = \frac{1}{2}A^{-\frac{1}{n}}\left(\frac{U}{L}\right)^{\frac{1-n}{n}}.$$

We set

$$\widehat{s}(\widehat{x}) = -\frac{\widehat{x}\tan(\alpha)}{\delta}, \qquad \widehat{b}(\widehat{x}) = \widehat{s}(\widehat{x}) - 1 + \frac{1}{2}\sin(2\pi\widehat{x}) \quad (36)$$

with $\alpha = 0.5°$ and $\delta = \frac{1}{80}$. The exact solution for the velocity and pressure is defined to be $u = U\widehat{u}$, $w = W\widehat{w}$, and $p = P\widehat{p}$, where

$$\begin{cases} \widehat{u}(\widehat{x},\widehat{z}) &= \dfrac{c_x}{1-\frac{1}{2}\sin(2\pi\widehat{x})}\left[1-\left(\dfrac{-\widehat{z}-\widehat{x}\tan(\alpha)}{1-\frac{1}{2}\sin(2\pi\widehat{x})}\right)^{\lambda}\right], \\ \widehat{w}(\widehat{x},\widehat{z}) &= \widehat{u}(\widehat{x},\widehat{z})\left(\dfrac{d\widehat{b}}{d\widehat{x}}\dfrac{\widehat{s}-\widehat{z}}{\widehat{s}-\widehat{b}} + \dfrac{d\widehat{s}}{d\widehat{x}}\dfrac{\widehat{z}-\widehat{b}}{\widehat{s}-\widehat{b}}\right), \\ \widehat{p}(\widehat{x},\widehat{z}) &= 2\widehat{\mu}\dfrac{\partial\widehat{u}}{\partial\widehat{x}} - (\widehat{s}-\widehat{z}) \end{cases} \quad (37)$$

with $\lambda = 1$ and $c_x = 10^{-6}$.

The constructed velocity components satisfy the continuity equation (2). However, the velocity and pressure fields do not necessarily satisfy the conservation of momentum equations (1) nor the top-surface boundary condition (11) with $p_{atm} = 0$. Compensatory stresses are applied to make these equations hold. For the momentum equations (1), we define the compensatory stresses $\Sigma_x = \dfrac{P}{Z}\widehat{\Sigma}_x$, $\Sigma_z = \dfrac{P}{Z}\widehat{\Sigma}_z$, where

$$\begin{cases} \widehat{\Sigma}_x &= \delta\dfrac{\partial\left(2\widehat{\mu}\frac{\partial\widehat{u}}{\partial\widehat{x}}+\widehat{p}\right)}{\partial\widehat{x}} + \dfrac{\partial\left(\widehat{\mu}\left(\frac{1}{\delta}\frac{\partial\widehat{u}}{\partial\widehat{z}}+\delta\frac{\partial\widehat{w}}{\partial\widehat{x}}\right)\right)}{\partial\widehat{z}}, \\ \widehat{\Sigma}_z &= \delta\dfrac{\partial\left(\widehat{\mu}\left(\delta\frac{\partial\widehat{w}}{\partial\widehat{x}}+\frac{1}{\delta}\frac{\partial\widehat{u}}{\partial\widehat{z}}\right)\right)}{\partial\widehat{x}} + \dfrac{\partial\left(2\widehat{\mu}\frac{\partial\widehat{w}}{\partial\widehat{z}}+\widehat{p}\right)}{\partial\widehat{z}} - 1. \end{cases} \quad (38)$$

At the top surface, we define the compensatory stresses $\tau_x = P\widehat{\tau}_x$, $\tau_z = P\widehat{\tau}_z$, where

$$\begin{cases} \widehat{\tau}_x &= \dfrac{1}{\sqrt{1+\delta^2\left(\frac{d\widehat{s}}{d\widehat{x}}\right)^2}}\left[-\delta\dfrac{d\widehat{s}}{d\widehat{x}}\left(2\widehat{\mu}\dfrac{\partial\widehat{u}}{\partial\widehat{x}}+\widehat{p}\right)+\widehat{\mu}\left(\dfrac{1}{\delta}\dfrac{\partial\widehat{u}}{\partial\widehat{z}}+\delta\dfrac{\partial\widehat{w}}{\partial\widehat{x}}\right)\right], \\ \widehat{\tau}_z &= \dfrac{1}{\sqrt{1+\delta^2\left(\frac{d\widehat{s}}{d\widehat{x}}\right)^2}}\left[-\delta\dfrac{d\widehat{s}}{d\widehat{x}}\left(\widehat{\mu}\left(\delta\dfrac{\partial\widehat{w}}{\partial\widehat{x}}+\dfrac{1}{\delta}\dfrac{\partial\widehat{u}}{\partial\widehat{z}}\right)\right)+\left(2\widehat{\mu}\dfrac{\partial\widehat{w}}{\partial\widehat{z}}+\widehat{p}\right)\right]. \end{cases} \quad (39)$$

At the lateral boundaries, we impose periodic boundary conditions. At the ice-bedrock boundary, $u = U\widehat{u}$ and $w = W\widehat{w}$ are imposed.

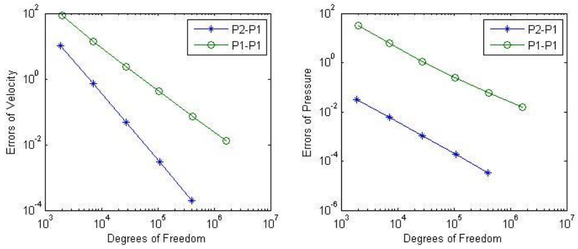

Figure 3. Model error for the analytical solution tests (L^2 norm). Left: velocity error; right: pressure error.

Table 1. Comparisons of errors in the velocity and pressure approximations by the high-order accurate P2-P1 finite element method and the low-order accurate P1-P1 finite element method.

Elements	Mesh	DOF	Velo. Error	Conv. Rate	Pres. Error	Conv. Rate
	20×3×5	1,894	1.01×10^{1}	-	3.25×10^{-2}	-
The P2-P1	40×3×10	7,144	7.27×10^{-1}	2.79	5.84×10^{-3}	1.48
	80×3×20	27,724	4.71×10^{-2}	2.95	1.04×10^{-3}	1.49
Finite element	160×3×40	109,204	3.04×10^{-3}	2.95	1.83×10^{-4}	1.50
	320×3×80	407,491	1.96×10^{-4}	2.96	3.24×10^{-5}	1.50
	20×3×5	2,016	8.79×10^{1}	-	4.25×10^{1}	-
The P1-P1	40×3×10	7,216	1.42×10^{1}	1.63	5.93×10^{0}	1.84
	80×3×20	27,216	2.39×10^{0}	1.57	1.11×10^{0}	1.41
finite element	160×3×40	105,616	4.15×10^{-1}	1.52	2.43×10^{-1}	1.20
	320×3×80	416,016	7.30×10^{-2}	1.51	6.07×10^{-2}	1.00
	640×3×160	1,651,216	1.29×10^{-2}	1.50	1.51×10^{-2}	1.01

Our Stokes solver applies to three-dimensional numerical ice sheet modeling. In order to generalize the two-dimensional problem above to three dimensions, we replicate the two-dimensional x, z-plane grid to 3 layers along the y-direction and impose periodic boundary conditions in y.

We ran this experiment on structured uniform grids using both our parallel high-order accurate P2-P1 finite element Stokes solver and the linear (penalized) P1-P1 finite element solver proposed in *Zhang et al.* [2011], which uses linear elements for both the velocity and pressure. The errors between the model and the analytical solution, measured by the L^2 norm, are reported in Table 1 and Figure 3. From these tables, we can see that the P2-P1 finite element scheme is much more accurate and efficient than the P1-P1 finite element method. For this test ice sheet example, the convergence rate (with respect to the grid size) of the velocity approximation using the P1-P1 element is only about 1.50, whereas that for the P2-P1 element is nearly 3.0 as expected. Thus, as the grid size is reduced, errors using the P2-P1 element reduce much faster than in the P1-P1 case. For example, in order to make the error of the velocity approximation less than 5×10^{-2} ma^{-1}, we need to use a $640 \times 3 \times 160$ grid ($1,651,216$ DOF) for the P1-P1 finite element method, but only a $80 \times 3 \times 20$ grid ($27,724$ DOF) for the P2-P1 element method. Note that no grid refinement is needed in the y direction because the exact two-dimensional solution, when extended to three dimensions, is independent of y.

4.2. ISMIP-HOM benchmark tests

Next, we test our parallel high-order accurate finite element nonlinear Stokes ice sheet model using the ISMIP-HOM benchmark problems [*Pattyn et al.*, 2008]. There are six experiments (Experiments A-F) in this benchmark suite. For all experiments, the horizontal extent of the ice sheet Ω_H is a rectangle and periodic boundary conditions are applied in at least one of the horizontal directions. As described in Section 3.1.1, we use tetrahedral grids produced from uniform two-dimensional structured triangular meshes. We use the parameters given in *Pattyn et al.* [2008] for the ice sheet; see Table 2.

Table 2. Parameters and constants for the ISMIP-HOM benchmark experiments for the nonlinear Stokes ice sheet model.

Name	Symbol	Value	Units
Deformation rate factor	A	10^{-16} (Exp. A-E) 2.140373×10^{-17} (Exp. F)	$Pa^{-n}a^{-1}$
Power law exponent	n	3 (Exp. A-E) 1 (Exp. F)	−
Ice density	ρ	910	$kg\,m^{-3}$

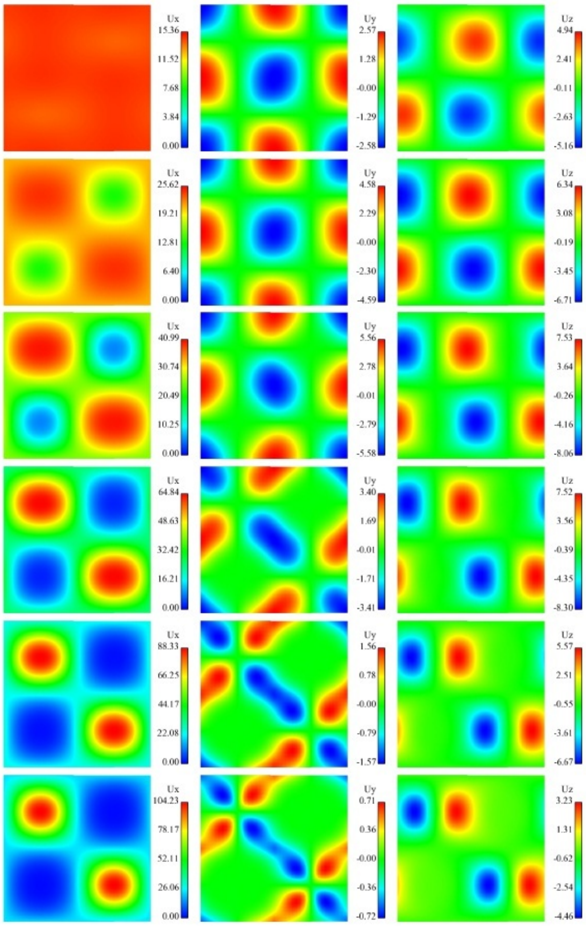

Figure 4. Simulation results at different length scales for Experiment A. From left to right: the components u_x, u_y, and u_z of the top surface velocity (ma^{-1}); from top to bottom: $L = 5$, 10, 20, 40, 80, 160 km.

4.2.1. Experiments A and B: ice flow with no-sliding basal boundary condition

Experiment A considers ice flow over a bumpy bed on an idealized geometry, i.e., we have a parallel-sided slab of ice with a mean thickness of 1000 m lying on a bed with a mean slope of $\alpha = 0.5°$. The basal topography of the ice sheet is then defined as a series of sinusoidal oscillations with an amplitude of 500 m:

$$z_b(x, y) = z_s(x) - 1000 + 500 \sin(\omega x) \sin(\omega y),$$

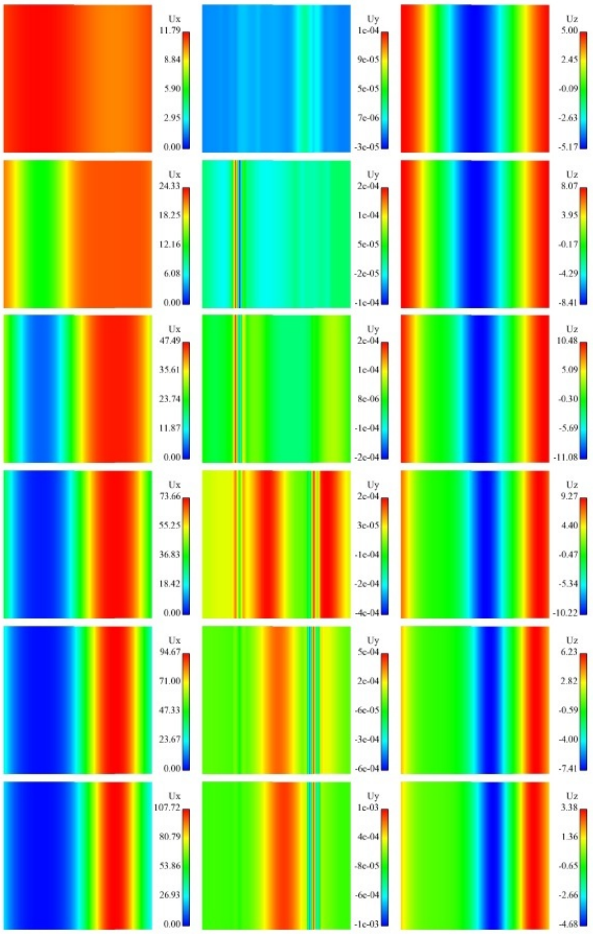

Figure 5. Simulation results at different length scales for Experiment B. From left to right: the components u_x, u_y, and u_z of the top surface velocity (ma^{-1}); from top to bottom: $L = 5$, 10, 20, 40, 80, 160 km.

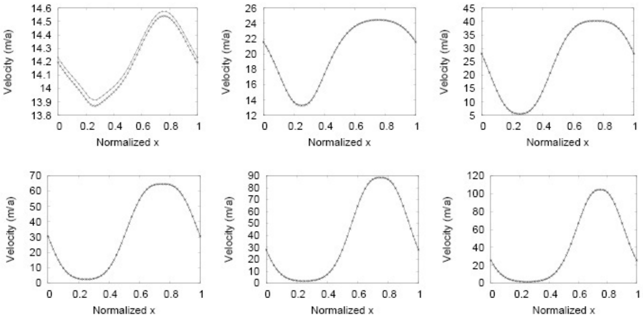

Figure 6. L^2-norm of the surface velocity across the bump at $y = L/4$ in Experiment A at different length scales (solid lines with tick marks: our solution; dashed curve with crosses: the reference solution [*Gagliardini et al.*, 2008]). From left to right and from top to bottom: $L = 5$, 10, 20, 40, 80, 160 km.

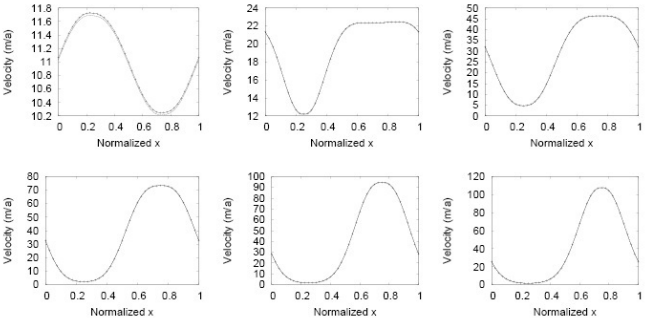

Figure 7. L^2-norm of the surface velocity across the bump at $y = L/4$ in Experiment B at different length scales (solid lines with tick marks: our solution; dashed curve: the reference solution [*Gagliardini et al.*, 2008]). From left to right and from top to bottom: $L = 5$, 10, 20, 40, 80, 160 km.

where the top surface is given by

$$z_s(x) = -x \tan\alpha. \tag{40}$$

Here, $(x, y) \in [0, L] \times [0, L]$ and the basal bumps have a frequency $\omega = 2\pi/L$. A no-sliding boundary condition $\mathbf{u} = \mathbf{0}$ is imposed at the bottom boundary of the ice sheet so that $\Gamma_{b,sld} = \emptyset$. On the lateral boundaries, periodic boundary conditions are imposed.

The set-up for Experiment B is the same except that the basal topography does not vary in y, i.e., we have a rippled bed given by

$$z_b(x, y) = z_s(x) - 1000 + 500\sin(\omega x),$$

where $z_s(x)$ is again given by (40).

The ice sheet length L is set to increase by a factor of two in each successive simulation, starting with $L = 5$ km and ending with $L = 160$ km. Note that for smaller domain aspect ratios (smaller values of L), the non-SIA parts of the model contribute relatively more to the stress balance, whereas for larger aspect ratios (larger values of L) the solution for the Stokes model is essentially the same as that for an SIA model [*Pattyn et al.*, 2008]. Our numerical simulation results for the surface velocity components at different length scales for Experiments A and B are shown in Figures 4 and 5, respectively; 16 processors and a grid of 192,000 tetrahedra resulting from a $40 \times 40 \times 20$ structured prismatic mesh was used. The L^2-norm of the surface velocity across the bump at $y = L/4$ is shown and compared with the reference solution from *Gagliardini et al.* [2008] in Figures 6 and 7. Our results match very well with the results of *Gagliardini et al.* [2008], also used a Stokes finite element model but with a finer $60 \times 60 \times 30$ mesh for Experiment A and a $240 \times 120 \times 30$ mesh for Experiment B.

4.2.2. Experiments C and D: ice flow with basal sliding condition

Experiments C and D consider the case of basal sliding conditions. The geometrical setup of the ice sheet in Experiment C is the same as that for Experiment A except that $\alpha = 0.1°$ and the basal topography is flat, i.e.,

$$z_b(x, y) = z_s(x) - 1000, \tag{41}$$

where $z_s(x)$ is again given by (40). Periodic boundary conditions are still imposed on the lateral boundaries, but on the bottom surface the basal friction boundary condition (13) with

$$\beta^2(x, y) = 1000 + 1000\sin(\omega x)\sin(\omega y). \tag{42}$$

is imposed everywhere so that $\Gamma_{b,fix} = \emptyset$.

The setup of Experiment D is the same as for Experiment C except that the basal friction coefficient is set to

$$\beta^2(x, y) = 1000 + 1000\sin(\omega x). \tag{43}$$

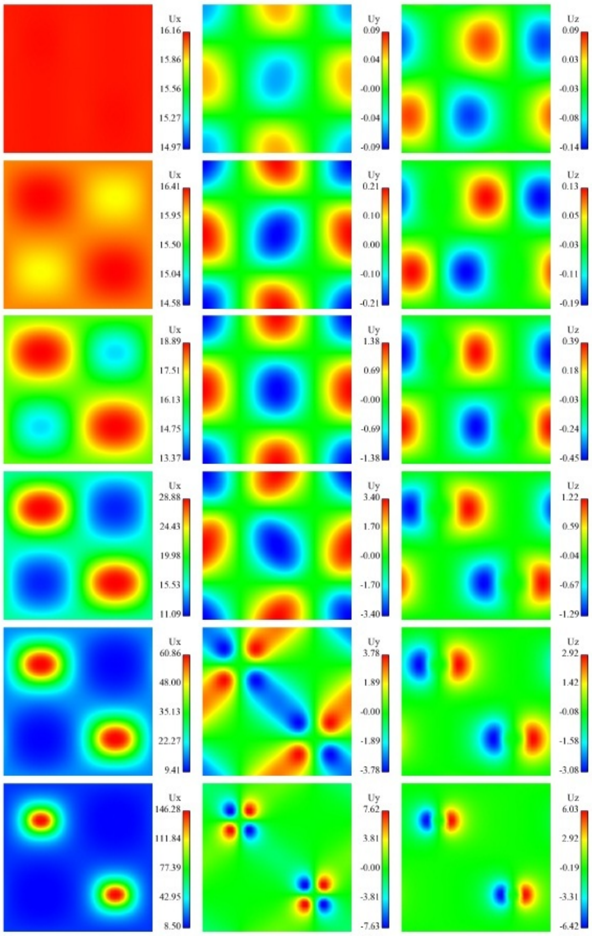

Figure 8. Simulation results at different length scales for Experiment C. From left to right: the components u_x, u_y, and u_z of the top surface velocity (ma^{-1}); from top to bottom: $L = 5$, 10, 20, 40, 80, 160 km.

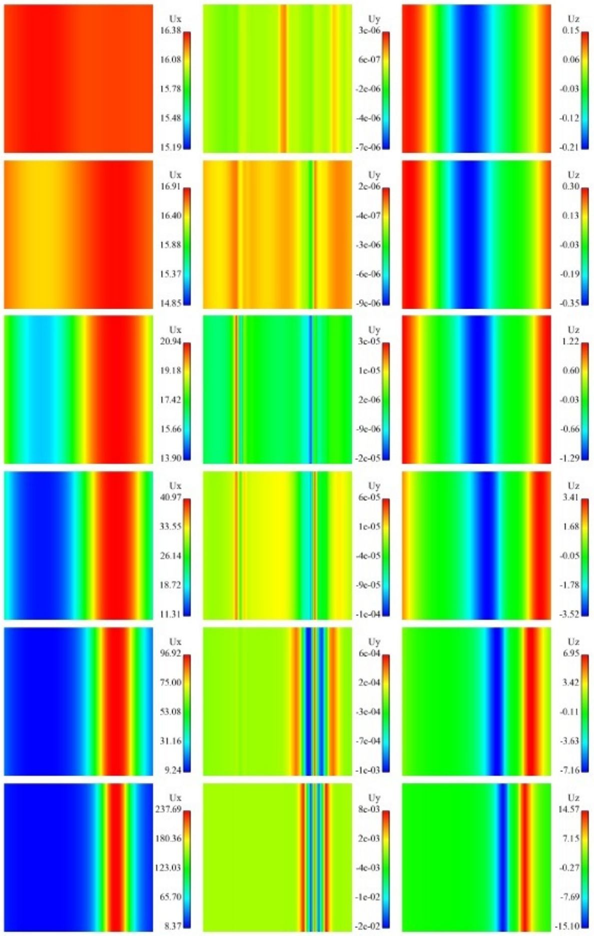

Figure 9. Simulation results at different length scales for Experiment D. From left to right: the components u_x, u_y, and u_z of the top surface velocity (ma^{-1}); from top to bottom: $L = 5$, 10, 20, 40, 80, 160 km.

Simulation results for the surface velocity components at different length scales for Experiments C and D are shown in Figures 8 and 9, respectively, using the same grid (except the elevation in the z direction) as that used for Experiments A and B. The L^2-norm of surface velocity across the bump at $y = L/4$ is shown and compared with the reference solution in Figures 10 and 11. Again, our results compare well with the results in *Gagliardini et al.* [2008].

4.2.3. Experiment E: Haut Glacier d'Arolla

Experiment E is a diagnostic experiment along the central flowline of a temperate glacier in the Alps (the Haut Glacier d'Arolla in Switzerland). The basic experiment and geometry is described in *Blatter et al.* [1998]. Input for the model includes the longitudinal surface and bedrock profiles of the Haut Glacier d'Arolla.

We run Experiment E for two different basal boundary conditions: (I) without sliding, i.e., the zero velocity condition $\mathbf{u} = 0$ (or $\beta^2 = +\infty$) is imposed on the basal boundary; (II) with a zone of sliding defined by

$$\beta^2 = \begin{cases} 0 & \text{if} \quad 2200m \le x \le 2500\text{m}, \\ +\infty & \text{otherwise}. \end{cases} \tag{44}$$

For both cases, zero velocity conditions are imposed on the lateral boundary in the x direction and periodic boundary conditions are imposed in the y direction.

Two different grids are generated using the data set for the Haut Glacier d'Arolla longitudinal profile, which consists of 51 bedrock and surface elevation points. The first grid, denoted by E_1_10, uses only one interval between the data set points of the geometry of the profile, whereas the second grid uses 10 intervals, denoted by E_10_10. Both grids have 10 layers in the vertical direction. The geometry of the glacier between neighboring data set points is not known, so we use linear interpolation to approximate mesh vertices at intervening points. The shape determined by linear interpolation is not very smooth, which might be realistic in some cases. For other applications, as suggested in *Gagliardini et al.* [2008], smoother interpolation such as nonuniform rational B-splines (NURBS) could be used instead.

Simulation results of Arolla flow without the sliding zone are shown in Figures 12 (left) and 13. The surface velocities for both grids are quite smooth and show good agreement with the reference solution, which uses a 4 times finer grid than E_10_10, but the basal shear stresses and the pressure differences show certain oscillations and roughly match the reference solution, especially for the coarse grid case. Clearly one interval between two data points is not enough and higher resolution is needed to capture all of the relevant features. The solution on the fine grid E_10_10 is smooth between two neighbor data points; however, near the data points sudden changes still appear. We can conclude that the solutions, especially of the high-order accurate numerical models such as the model proposed here, are quite sensitive to the geometry; rough boundary descriptions such as linear interpolation on the boundary could lead to solutions with small oscillations. When complex realistic ice

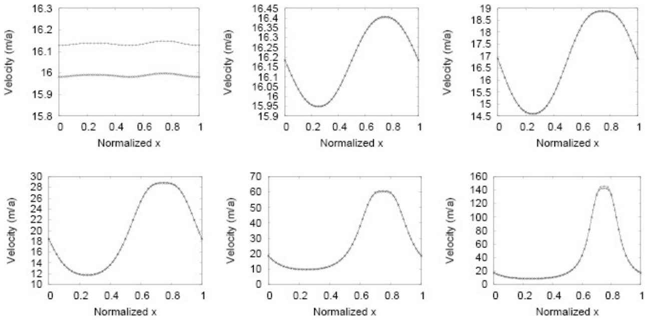

Figure 10. L^2-norm of the surface velocity across the bump at y = L/4 in Experiment C at different length scales (solid lines with tick marks: our solution; dashed curve with crosses: the reference solution [*Gagliardini et al.*, 2008]). From left to right and from top to bottom: $L = 5$, 10, 20, 40, 80, 160 km.

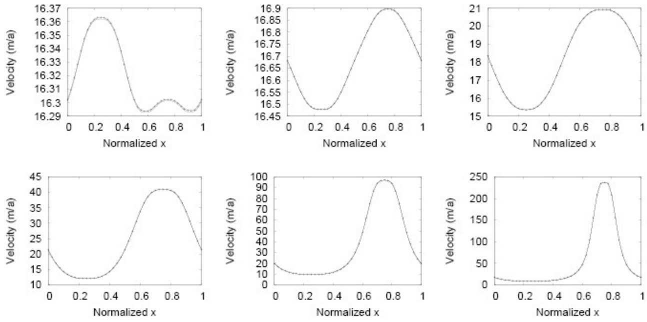

Figure 11. L^2-norm of the surface velocity across the bump at y = L/4 in Experiment D at different length scales (solid lines with tick marks: our solution; dashed curve: the reference solution [*Gagliardini et al.*, 2008]). From left to right and from top to bottom: $L = 5$, 10, 20, 40, 80, 160 km.

sheets such as the Greenland ice sheet are considered, because their surface elevation and bed topography may not be sufficiently smooth (or is not sampled densely enough), use of coarse boundary descriptions and high-order numerical models could cause unstable oscillations at places where the geometry is not smooth or has sharp changes. In that case, we suggest using adaptive mesh refinement in these locations or the use of NURBS during interpolation of data onto the model mesh.

The results of Arolla flow with a sliding zone are shown in Figures 12 (right) and 14. Because this experiment includes a sliding zone, a singularity appears at the points where friction parameter β^2 suddenly changes from zero to infinity. The results using the coarse grid E_1_10 are inaccurate; as we can see, the velocity is relatively smaller and the peak basal stress and pressure difference are smoothed out. For the results using the fine grid E_10_10, the strong singularities of the basal stress and the pressure difference near the singular points are more pronounced, but in smooth regions our results show, for the surface velocity, agreement with the reference solution from *Gagliardini et al.* [2008] that is determined using a grid that is four times finer. In general, with the types of finite element functions used in this work, the singular behavior cannot be fully represented in simple ways such as using local mesh refinement, although that can help somewhat. Whereas the singularity in this sample problem is somewhat unphysical, ongoing work will focus on how better to model rapid and/or short-length scale switches in basal boundary conditions.

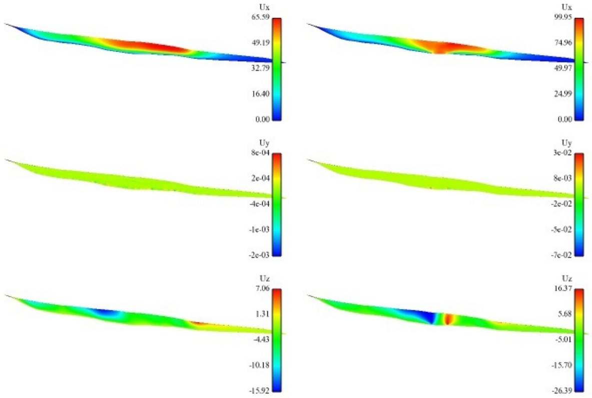

Figure 12. Simulation results (view of the middle section along the y-direction) for Experiment E. Left: Case I – without sliding zone; right: Case II – with sliding zone. From top to bottom: the components u_x, u_y, u_z of the velocity.

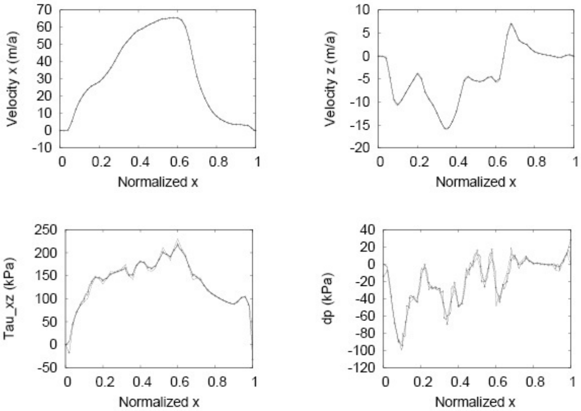

Figure 13. Simulation results for Experiment E without a sliding zone (Case I). From left to right then from top to bottom: horizontal top surface velocity (ma^{-1}); vertical top surface velocity (ma^{-1}); basal shear stress; difference between the isotropic and hydrostatic pressure at the bed. Dotted curve: fine mesh; dashed curve with crosses: coarse grid; solid curve with tick marks: the reference solution [*Gagliardini et al.*, 2008].

4.2.4. Experiment F: a prognostic example

Experiment F is a prognostic example for which the top surface is allowed to evolve in time according to equation (10) until a steady state is reached such that

$$\lim_{t\to+\infty} \frac{\partial z_s}{\partial t} = 0. \qquad (45)$$

A slab of ice with mean ice thickness $H^{(0)} = 1000\,\mathrm{m}$ is considered, lying on a bed with a mean slope of $\alpha = 3.0°$. This slope is maximum in x and zero in y. The initial top surface elevation is

$$z_s^{(0)}(x,y) = -x\tan\alpha \qquad (46)$$

and the bedrock plane z_b is parallel to the initial top surface plane but is perturbed by a Gaussian bump such that

$$z_b(x,y) = -H^{(0)} - x\tan\alpha + \gamma_0 e^{-(x^2+y^2)/\sigma^2} \qquad (47)$$

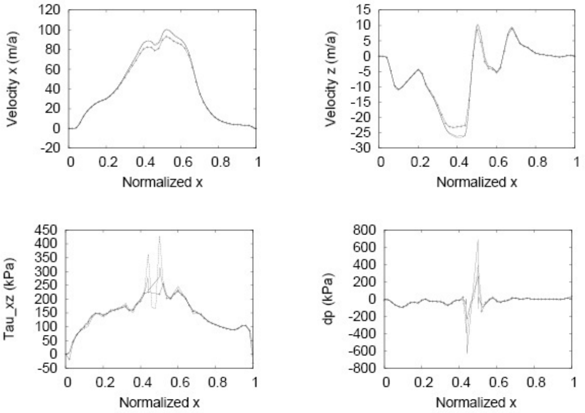

Figure 14. Simulation results of Experiment E with a sliding zone (Case II). From left to right then from top to bottom: horizontal top surface velocity (ma^{-1}); vertical top surface velocity (ma^{-1}); basal shear stress; difference between the isotropic and hydrostatic pressure at the bed. Dotted curve: fine mesh; dashed curve with crosses: coarse grid; solid curve with tick marks: the reference solution [*Gagliardini et al.*, 2008].

with $\gamma = 0.1H^{(0)}$ and $\sigma = 10H^{(0)}$. The horizontal domain of the ice sheet is taken to be a square $[-L/2, L/2] \times [-L/2, L/2]$ with size $L = 100H^{(0)}$. It is also assumed that there is no surface accumulation/ablation so that the initial Gaussian surface bump is allowed to relax over time. Periodic boundary conditions are applied in the horizontal directions and a sliding boundary condition is imposed on the bottom with

$$\beta^2 = (cAH^{(0)})^{-1}. \tag{48}$$

We test two cases for Experiment F: (I) a non-slippery case, i.e., $c = 0$, and (II) a slippery case with $c = 1$. Note that in this experiment, because $n = 1$, the effective viscosity is constant and is given by $\eta_{\mathbf{u}} \equiv (2A)^{-1}$ so that the Picard iteration is not needed.

The time-dependent model is run using our parallel implementation, again on the same tetrahedral grids (except for the elevation in the z direction) as for Experiments A-D with a time step of 5 years. We also note the z-coordinates of the grid points are recalculated at each time step to maintain the mesh quality as mentioned in Section 3.1.1. In our test, a steady state is assumed to be reached if the change of the top surface elevation between two consecutive time steps is small enough, e.g., less than 10^{-2} m. From our simulation, the time it takes for the thickness of the ice sheet to be stabilized in the non-slippery case and slippery case are about 375 and 1245 years, respectively. The steady state results and convergence history are shown in Figures 15–17. Our results show good agreement with the reference solutions in *Gagliardini et al.* [2008]; *Martin et al.* [2003].

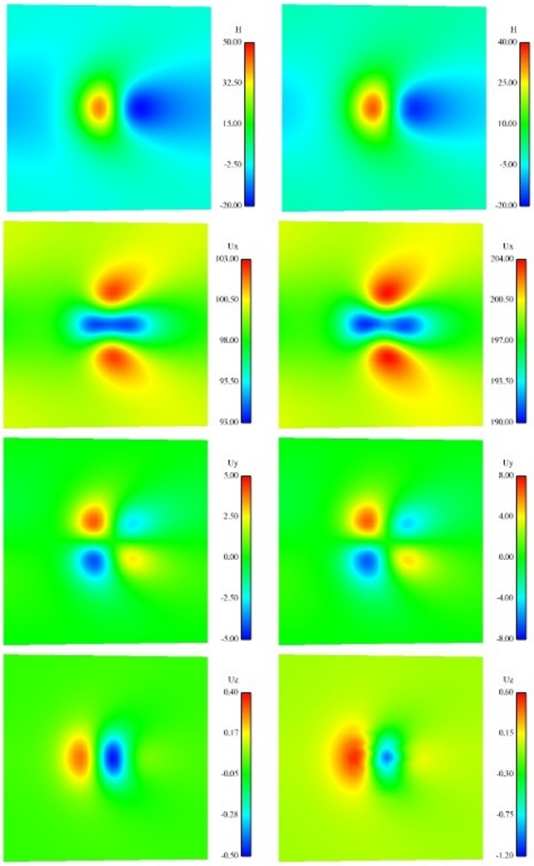

Figure 15. Simulation results for the steady state in Experiment F viewed from above the top surface. Left: Case I – non-slippery case; right: Case II – slippery case. Top to bottom: surface elevation and velocity components u_x, u_y, and u_z.

5. Performance and scalability

We now summarize the computational performance of our parallel high-order accurate finite element Stokes model, which performed well in all of the experiments discussed above. We first present tests of the linear system solver strategy we use. As described in Section 3.1.1, a grid of 192,000 tetrahedra was produced based on a structured prismatic mesh for a 40 × 40 × 20 decomposition of the domain; the resulting number of degrees of freedom for the discretized system is 827,604. Three different AMG parameter settings were used, as listed in Table 3; the first setting was also used in *Burstedde et al.* [2009]; *May and Moresi* [2008]. The predetermined settings of the AMG preconditioner and numbers of the inner GMRES iterations (NOIs-innerG) used in Experiments A-D are listed in Table 4. We note that the selections are based on our experiences (the relative residuals roughly reduce to 10^{-1}). The number of the outer FGMRES iterations of the linear system solver on the mesh of size 40 × 40 × 20 are listed in Table 5.

From the table it is easy to see that our linear solver worked very well for all experiments, i.e., it used only a few outer FGMRES iterations. It is also clear that problems with sliding boundary conditions are more difficult than the ones with a no-sliding boundary condition and require 3 to 6 times the number of inner GMRES iterations to effect similar reductions in the residual of the subproblem solver. This is usually due to the linear system being more poorly conditioned for sliding than for no-sliding basal boundary conditions.

In parallel computing, two types of scalabilities are commonly used to measure the performance. The first one is strong scalability S_{strong}, which reflects how the solution time varies with the number of processors for a fixed problem size. The second is weak scalability S_{weak}, which shows how the solution time varies with the number of processors for a fixed problem size per processor. Because we used four processors as the smallest number of processors in all tests,

Table 3. The BoomerAMG parameter setting groups used in our parallel solver.

Parameter Setting Group	a	b	c
Coarsening	PMIS	PMIS	Falgout
Interpolation	extended	extended	classical
Truncation factor	0.3	0.3	0
Strong threshold	0.5	0.9	0.5
Max entries per row for interp	5	5	0
Number of Functions	3	3	1
Coarsest Relax Type	direct	direct	direct

Table 4. The predetermined AMG setting groups (defined in Table 3) and numbers of the inner GMRES iterations for solving the subproblem used in Experiments A-D.

Length Scale	Exp. A		Exp. B		Exp. C		Exp. D	
	AMG Setting	NOIs-innerG	AMG Setting	NOIs-innerG	AMG Setting	NOIs-innerG	AMG Setting	NOIs-innerG
$L = 5$km	a	1	a	1	c	3	c	6
$L = 10$km	a	1	a	1	c	3	c	6
$L = 20$km	a	1	a	1	c	3	c	6
$L = 40$km	a	1	b	1	c	3	c	6
$L = 80$km	a	1	b	1	c	3	c	6
$L = 160$km	a	1	b	1	c	3	c	6

Table 5. Numbers of the outer FGMRES iterations of the preconditioned solver on the mesh of size $40 \times 40 \times 20$ in Experiments A-D.

Length Scale	Exp. A	Exp. B	Exp. C	Exp. D
$L = 5km$	27	46	61	120
$L = 10km$	26	37	34	76
$L = 20km$	24	48	23	58
$L = 40km$	30	49	15	56
$L = 80km$	48	48	16	60
$L = 160km$	50	47	28	103

Table 6. Results of the weak scalability tests using our parallel FEM solver for Experiment A with $L = 80$ km.

Mesh Size	Number of Tetrahedra	DOF	Number of Procs	NOIs-outerFG	NOIs-nonlinear	Total Time(s)	Scalability S_{weak}
20 × 20 × 20	48,000	208,644	4	62	25	622	-
40 × 40 × 20	192,000	827,604	16	51	25	598	1.040
80 × 80 × 20	768,000	3,296,724	64	33	25	570	1.091
160 × 160 × 20	3,072,000	13,159,764	256	32	25	694	0.896
320 × 320 × 20	12,288,000	52,585,044	1024	34	25	1146	0.543

Table 7. Results of the strong scalability tests using our parallel FEM solver for Experiment A with $L = 80$ km. Mesh size: 80 × 80 × 20, number of tetrahedral elements: 768,000, number of DOF: 3,296,724.

Number of Processors	Memory Usage per Core(MB)	Average Time(s) per Picard Ite.	Total Time(s)	Scalability S_{strong}
4	5058	301	8778	-
8	2653	126	3702	1.186
16	1422	63	1857	1.182
32	823	33	997	1.101
64	487	18	526	1.043
128	339	10	301	0.911
256	267	8	236	0.775

we calculated S_{strong} and S_{weak} as

$$S_{strong} = \frac{4T_4}{nT_n}, \qquad S_{weak} = \frac{T_4}{\widehat{T}_n}, \tag{49}$$

where T_n denotes the solution time with n processors when the problem size is fixed and $\widehat{T}_n$ the solution time with n processors when the problem size is similarly scaled along with the number of processors. A value of S_{strong} or S_{weak} greater than 1.0 implies super-linear scaling.

The first test is for weak scalability, i.e., we test our solver with proportionally increasing number of processors but a fixed subproblem size per processor. Up to 1024 processors are used to test our solver using Experiment A with $L = 80$ km and the results are reported in Table 6. In this table, ”NOIs-nonlinear” denotes the number of Picard iterations and “NOIs-outerFG” the number of outer FGMRES iterations used for solving the linear system at each Picard iteration. From the table we can see that the number of iterations the linear solver used for different problem sizes is almost constant, and the total running times grow very slow as the number of processors increases until the last case with 1024 processors. This clearly shows both our preconditioning algorithm and parallel implementation result in good weak scalability.

The second test is for strong scalability, i.e., we test our solver with increasing number of processors for a fixed total problem size. We fixed the structured mesh using the 80 × 80 × 20 decomposition of the domain; the number of DOF is 3,296,724. Results are reported in Table 7. For all

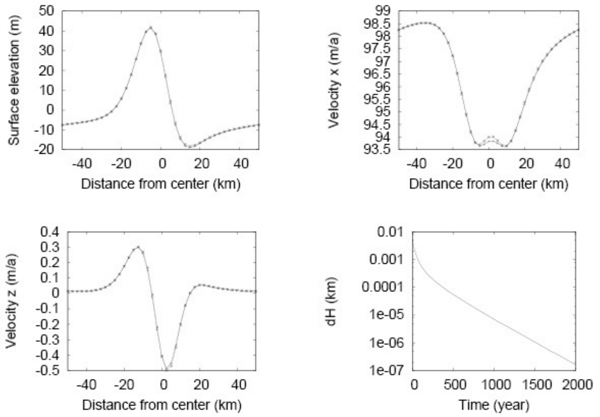

Figure 16. Simulation results for the steady state in Experiment F – Case I (non-slippery case) viewed along the central line. From left to right then from top to bottom: top surface elevation; surface velocity component u_x; surface velocity component u_z; maximum change of the surface elevation in time. Dotted curve with crosses: our solution; dashed curve with crosses: the reference solution [*Martin et al.*, 2003].

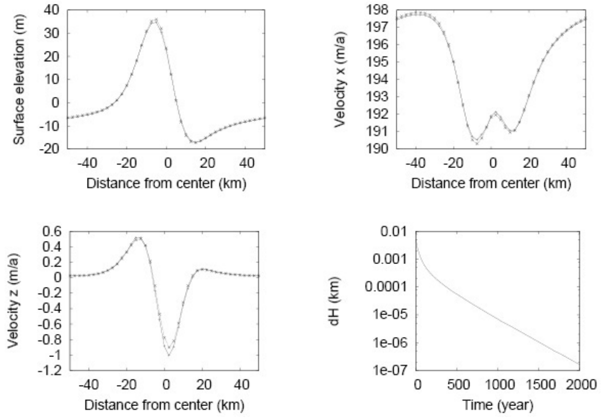

Figure 17. Simulation results of the steady state in Experiment F – Case II (slippery case) viewed along the central line. From left to right then from top to bottom: top surface elevation; surface velocity component u_x; surface velocity component u_z; maximum change of the surface elevation in time. Dotted curve with crosses: our solution; dashed curve with crosses: the reference solution [*Martin et al.*, 2003].

the tests with different number of processors, the number of outer FGMRES iterations is 33 and the number of nonlinear (Picard) iterations is 25. From the table it is easy see that, when the number of processors is no more than 32, the total computing times decrease by more than a factor of two as the number of processors doubles, implying a super linear scaling. For the rest of the cases, the computing time decrease is not so large but is still significant. This shows that our parallel solver has excellent strong scalability as well.

6. Concluding remarks

In this paper, we developed a parallel, three-dimensional, finite element ice sheet model based on the nonlinear Stokes system. Our finite element model features high-order computational accuracy and also features scalable parallel solvers.

Future efforts will be devoted to implementing the three-dimensional nonlinear Stokes finite element model on variable resolution grids and in particular, for modeling of the Greenland and Antarctica ice sheets. Note that our tetrahedral meshing strategy based on prismatic grids is directly applicable to grids constructed using variable resolution in the horizontal directions and layers in the vertical direction. Future efforts will also focus on energetics coupling, the implementation of other sliding/friction basal laws (such as Coulomb friction), and treatment of the ice-ocean interface. We will also examine methods to ameliorate the singularity that accompanies sudden transitions from no-sliding to sliding/friction basal boundary conditions. Finally, we will explore several computational implementation improvements for our ice sheet model, including the use of Newton's method for accelerating the nonlinear solution process, the impact of using adaptive NURBS refined elements in Experiments E and F to produce better and smoother solutions, and designing better preconditioning strategies for problems with sliding boundary conditions to produce more efficient solvers.

Acknowledgments. We thank the editor Alexander Densmore, the associated editor Martin Truffer, as well as Dan Goldberg, David Egholm, and Eric Larour for thorough and insightful reviews that helped to substantially improved the paper. This work was partially supported by the US DOE Office of Science Climate Change Prediction Program through DE-FG02-07ER64431, DE-FG02-07ER64432, and DOE 07SCPF152 and by the US National Science Foundation under grant number DMS-0913491.

References

Alley, R.B., P.U. Clark, P. Huybrechts, and I. Joughin (2005), Ice sheet and sea-level changes, *Science*, 310, 456-460.

Balay S., K. Buschelman, V. Eijkhout, W.D. Gropp, D. Kaushik, M.G. Knepley, L.C. McInnes, B.F. Smith, and H. Zhang (2004), *PETSc users manual*, Technical Report ANL-95/11 - Revision 2.1.5, Argonne National Laboratory.

Bamber, J.L., R.B. Alley, and I. Joughin (2007), Rapid response of modern day ice sheets to external forcing, *Earth Planet Sci. Lett.*, 257, 1-13.

Bamber, J.L., S. Ekholm, and W. Krabill (2001), A new, high resolution digital elevation model of Greenland fully validated with airborne laser altimeter data, *J. Geophys. Res.*, 106 (B4), 6733-6745.

Bercovier, M., and O. Pironneau (1979), Error estimates for finite element method solution of the Stokes problem in the primitive variables, *Numer. Math.*, 33, 211-224.

Blatter, H. (1995), Velocity and stress fields in grounded glaciers: a simple algorithm for including deviatoric stress gradients, *J. Glaciol.* , 41, 33-44.

Blatter, H., G. Clarke, and J. Colinge (1998), Stress and velocity fields in glaciers: Part II. sliding and basal stress distribution., *J. Glaciol.*, 44, 457-466.

Briggs, W.L., V.E. Henson, and S.F. McCormick (2000), *A Multigrid Tutorial*, 2nd edition, SIAM.

Bromwich, D.H., L.H. Bai, and G.G. Bjarnason (2005), High-resolution regional climate simulations over Iceland using Polar MM5, *Monthly Weather Review*, 133(12), 3527-3547.

Burstedde, C., O. Ghattas, G. Stadler, T. Tu, and L.C. Wilcox (2009), Parallel scalable adjoint-based adaptive solution of variable-viscosity Stokes flow problems, *Comput. Methods Appl. Mech. Engrg.*, 198, 1691-1700.

Douglas, C.C. (1996), Multigrid methods in science and engineering, *IEEE Computational Science & Engineering*, 3(4), 55-68.

Dukowicz, J., S.F. Price, and W.H. Lipscomb (2011), Incorporating arbitrary basal topography in variational formulation of ice sheet models, *J. Glaciol.*, 57, 461-467.

Durand, G., O. Gagliardini, B. Fleurian, T. Zwinger, and E. Le Meur (2009), Marine ice sheet dynamics: Hysteresis and neutral equilibrium, *J. Geophys. Res.*, 114, (F03009), doi:10.1029/2008JF001170.

Elman H.C., D.J. Silvester, and A.J. Wathen (2005), *Finite Elements and Fast Iterative Solvers: With Applications in Incompressible Fluid Dynamics*, Oxford University Press.

Falgout, R.D., and U.M. Yang (2002), Hypre: A library of high performance preconditioners, *Proceedings of the International Conference on Computational Science-Part III*, 2331, 632-641.

Frayssé, V., L. Giraud, and S. Gratton (1998), A set of flexible GMRES routines for real and complex arithmetics, *CERFACS Technical Report TR/PA/98/20*, CERFACS, Toulouse, France.

Gagliardini, O., and T. Zwinger (2008), The ISMIP-HOM benchmark experiments performed using the finite element code Elmer, *The Cryosphere*, 2, 67-76.

Goldberg, D., D.M. Holland, and C. Schoof, C.. (2009), Grounding line movement and ice shelf buttressing in marine ice sheets, *J. Geophys. Res.*, 115, (F04026), doi:10.1029/2008JF001227.

Gregory, J.M., and P. Huybrechts (2006), Ice sheet contributions to future sea-level change, *Proc. R. Soc. A*, 364, 1709-1731.

Gunzburger, M. (1989), *Finite Element Methods for Viscous Incompressible Flows*, Academic, Boston.

Hindmarsh, R.C.A. (2004), A numerical comparison of approximations to the Stokes equations used in ice sheet and glacier modeling, *J. Geophys. Res.*, 109, (F01012), doi:10.1029/2003JF000065.

Holland, D.M., R.G. Ingram, L.A. Mysak, and J.M. Oberhuber (1995), A numerical simulation of the sea-ice cover in the Northern Greenland sea, *J. Geophys. Res.*, 100(C3), 4751-4760.

Holland, D.M., L.A. Mysak, D.K. Manak, and J.M. Oberhuber (1993), Sensitivity study of a dynamic thermodynamic sea ice model, *J. Geophys. Res.*, 98 (C2), 2561-2586.

Ju, L. (2007), Conforming centroidal Voronoi Delaunay triangulation for quality mesh generation, *Inter. J. of Numer. Anal. Model.*, 4, 531-547.

Karypis, G. (2003), Multi-constraint mesh partitioning for contact/impact computations, *Supercomputing 2003*, Phoenix, Arizona, USA.

Le Meur, E., O. Gagliardini, T. Zwinger, and J. Ruokolainen (2004), Glacier flow modelling: A comparison of the shallow ice approximation and the full-Stokes solution, *Comptes Rendus Physique*, 5(7), 709-722.

Li, X.Y., and J.W. Demmel (2003), SuperLU_DIST: A scalable distributed-memory sparse direct solver for unsymmetric linear systems, *ACM Trans. Math. Soft.*, 29(2), 110-140.

Martin, C., R.C.A. Hindmarsh, and F.J. Navarro. (2009), On the effects of divide migration, along-ridge flow, and basal sliding on isochrones near an ice divide, *J. Geophys. Res.*, 114, (F02006), doi:10.1029/2008JF001025.

Martin, C., F. Navarro, J. Otero, M. Cuadrado, and M. Corcuera (2003), Three-dimensional modelling of the dynamics of Johnson Glacier (Livingstone Island, Antarctica), *Ann. Glaciol.*, 39, 1-8.

May, D.A. and L. Moresi (2008), Preconditioned iterative methods for Stokes flow problems arising in computational geodynamics, *Phys. Earth Planet. Interiors*, 171 (1-4), 33-47, DOI:10.1016/j.pepi.2008.07.036.

Mernild, S.H., G.E. Liston, C.A. Hiemstra, and K. Steffen (2008), Surface melt area and water balance modeling on the Greenland ice sheet 1995-2005, *Journal of Hydrometeorology*, 9(6), 1191-1211.

Mernild, S.H., G.E. Liston, D.L. Kane, N.F. Knudsen, and B. Hasholt (2008), Snow, runoff, and mass balance modeling for the entire Mittivakkat Glacier (1998-2006), Ammassalik Island, SE Greenland, *Geografisk Tidsskrift-Danish Journal of Geography*, 108(1), 121-136.

Morlighem, M., E. Rignot, H. Seroussi, E. Larour, H.B. Dhia, and D. Aubry (2010), Spatial patterns of basal drag inferred using control methods for a full-Stokes and simpler models for Pine Island Glacier, West Antarctica, *J. Geophys. Res.*, 37, (L14502), doi:10.1029/2010GL043853.

MSC Software (2009), Nastran Product Information & Documentation, Documentation ID: DOC9282, http://simcompanion.mscsoftware.com/.

Neumann, T.A., H. Conway, S. Price, E.D. Waddington, G.A. Catania, and D.L. Morse (2008), Holocene accumulation and ice sheet dynamics in central West Antarctica, *J. Geophys. Res.*, 113, (F02018), doi:10.1029/2007JF000764.

Nick, F.M., A. Vieli, I.M. Howat, and I. Joughin (2009), Large-scale changes in Greenland outlet glacier dynamics triggered at the terminus, *Nature Geoscience*, 2(2), 110-114.

Nye, J. (1957), The distribution of stress and velocity in glaciers and ice sheets, *Proc. R. Soc. London*, Ser. A, 239, 113-133.

Olshanskii, M.A., J. Peters, and A. Reusken (2006), Uniform preconditioners for a parameter dependent saddle point problem with application to generalized Stokes interface equations, *Numer. Math.*, 105(1), 159-191.

Olshanskii, M.A., and A. Reusken(2006), Analysis of a Stokes interface problem, *Numerische Mathematik*, 103(1), 129-149.

Paterson, W. (1994), *The Physics of Glaciers*, Elsevier Science, Oxford, United Kingdom.

Pattyn, F. (2003), A new 3D higher-order thermomechanical ice sheet model: Basic sensitivity, ice-stream development and ice flow across subglacial lakes, *J. Geophys. Res.*, 108(B8), 2382, doi:10.1029/2002JB002329, 2003.

Pattyn, F., L. Perichon, A. Aschwanden, B. Breuer, D.B. Smedt, O. Gagliardini, G.H. Gudmundsson, R.C.A. Hindmarsh, A. Hubbard, J.V. Johnson, T. Kleiner, Y. Konovalov, C. Martin, A.J. Payne, D. Pollard, S. Price, M. Ruckamp, F. Saito, S. Sugiyama, S., and T. Zwinger (2008), Benchmark experiments for higher-order and full-Stokes ice sheet models (ISMIP-HOM), *The Cryosphere*, 2, 95-108.

Plummer, J., C. van der Veen (2011), A high resolution bed elevation map for Jakobshavn Isbrae, West Greenland, *preprint*. Also see https://www.cresis.ku.edu/∼ plummer/jakob.html.

Price, S.F., H. Conway, and E.D. Waddington (2007), Evidence for late Pleistocene thinning of Siple Dome, West Antarctica, *J. Geophys. Res.*, 112, F03021, doi:10.1029/2006JF000725.

Sadd, Y. (1993), A flexible inner-outer preconditioned GMRES alsgorithm, *SIAM J. Sci. Comput.*, 14, 461-469.

Saito, F. (2002), *Development of a Three Dimensional Ice Sheet Model for Numerical Studies of Antarctic and Greenland Ice Sheets*, Ph.D. thesis, Univ. of Tokyo, Tokyo, Japan.

Sargent, A., and J.L. Fastook (2010), Manufactured analytical solutions for isothermal full-Stokes ice sheet models, *The Cryosphere*, 4, 285-311.

Schoof, C. (2006), Variational methods for glacier flow over plastic till, *J. Fluid Mech.*, 555, 299-320.

Schoof, C. (2007), Ice sheet grounding line dynamics: steady states, stability, and hysteresis, *J. Geophys. Res.*, 112, (F03S28), doi:10.1029/2006JF000664.

Schoof, C. (2010), Coulomb friction and other sliding laws in a higher order glacier flow model, *Math. Model. Meth. Appl. Sci.*, 20, 157-189.

Shepherd, A., and D. Wingham (2007), Recent sea-level contributions of the Greenland and West Antarctic ice sheets, *Science*, 315, 1529-1532.

Silvester, D., H. Elman, D. Kay, and A. Wathen (2001), Efficient preconditioning of the linearized Navier-Stokes equations for incompressible flow, *J. Comput. Appl. Math.*, 128, 261-279.

Truffer, M., K.A. Echelmeyer, W.D. and Harrison (2001), The implication of till on glacier motion, *J. Glaciol.*, 47, 123-134.

Truffer, M., and K.A. Echelmeyer (2003), Of isbrae and ice streams, *Ann. Glaciol.*, 36(1), 66-72.

Valli, A.Q. (1999), *Domain Decomposition Methods for Partial Differential Equations*, Oxford Science Publications.

Zhang, H., L. Ju, M. Gunzburger, T. Ringler, and S. Price (2011), Coupled models and parallel simulations for three dimensional full-Stokes ice sheet modeling, *Numer. Math. Theo. Meth. Appl.*, 4, 359-381.

Zwinger, T., R. Greve, O. Gagliardini, T. Shiraiwa, and M. Lyly (2007), A full Stokes-flow therm-mechanical model for firn and ice applied to the Gorshkov crater glacier, Kamchatka, *Ann. Glaciol.*, 45, 29-37.